GIOCHI FANTASTICI PER VERI COMPUTER.

PER COMMODORE VIC20 E CBM64, APPLE, ATARI, IBM ED ALTRI

Sirius è la grande casa americana che programma e produce i videogiochi che più hanno successo nel mondo. Brillanti e pieni di invenzioni, graficamente perfetti i videogiochi Sirius sono ora disponibili in una gamma di oltre 40 titoli.

DISTRIBUITI DA

sirius elettronica

QUANTI COLORI HA LA TUA STAMPANTE ?

NEL 1983 LA SEIKOSHA PER PRIMA AL MONDO
E' IN GRADO DI PRESENTARE LA NUOVA STAMPANTE
GRAFICA A SETTE COLORI.

RIUNITE IN UN APPARECCHIO PRATICO E COMPATTO
LE CARATTERISTICHE DELLA STAMPANTE E DEL PLOTTER,
LA SEIKOSHA INVENTA UN NUOVO TIPO DI PERIFERICA
CHE BEN PRESTO SARA' INSOSTITUIBILE.

REBIT COMPUTER E' ORGOGLIOSA DI LANCIARE
QUESTA NOVITA' ASSOLUTA SUL MERCATO ITALIANO
AD UN PREZZO MOLTO, MOLTO COMPETITIVO:
MENO DI UN MILIONE.
MENO DI UNA COMUNE STAMPANTE IN BIANCONERO.

REBIT COMPUTER
A DIVISION OF G.B.C.

GP-700A

Graphic Color Printer

SEIKOSHA

SOMMARIO

Commodore Computer Club - Rivista indipendente per gli utenti di sistemi Commodore.
Direttore responsabile: Michele di Pisa
Redazione: Alessandro De Simone.
Direzione, redazione: Piazza Arduino, 3 - 20149 Milano - Tel. (02) 434354-435717
Pubblicità: Milano Paola Bevilacqua, Gianluigi Centurelli, Tina Ronchetti, Villa Claudio - Viale Famagosta, 75 - 20142 Milano - tel. (02) 8467348/9/40
Prezzi e abbonamenti: la rivista esce bimestralmente. Prezzo per una copia Lire 2.000. Arretrati il doppio. Abbonamento per dieci fascicoli lire 18.000. Abbonamento annuo cumulativo alle riviste Computer e Commodore Computer. Club (tariffa riservata agli studenti) L. 24.000. I versamenti vanno indirizzati a Minisystems Italia s.r.l., mediante assegno bancario, vaglia o utilizzando il c/c postale n. 11909207.
Composizioni: Minisystems Italia
Selezioni: Org. Aldo Ghiacci
Stampa: La Litografica s.r.l. - Busto A.
Registrazione: Tribunale di Milano n. 370 del 2/10/1982 - Sped. in abb. post. gr. III n. 70 quale supplemento alla rivista Computer - Pubbl. inferiore al 70%

Questo "executive" me lo rubo

Al recente Smau, il noto comico e cabarettista Francesco Salvi, il 64 in versione Executive voleva quasi rubarlo. Ora sta preparando un film il cui protagonista è proprio un computer Commodore.

FRANCESCO Salvi non è solo appassionato di informatica. Lui, per i computers ci va addirittura pazzo. E forse è per questo che è così abile e scaltro nel suo mestiere di comico e cabarettista. Forse perchè ha la lucidità mentale di un programmatore o forse perchè usa un Commodore per memorizzare battute (di spirito). O forse perchè è proprio pazzo.

Ad ogni modo, la sua carriera va a gonfie vele. Al Derby e negli altri locali in cui si fa cabaret è contesissimo; alla RAI Rete 2 sta andando in onda la trasmissione Galassia 2, dove Salvi è l'ospite fisso, e di certo contribuisce ad innalzare l'ascolto della trasmissione di Boncompagni. In progetto un altro film fra computer, terminali video e stampanti: uno

scienziato pazzo (vuoi il caso), che ha in mano il potere grazie all'utilizzo dell'informatica.

Alla recente manifestazione fieristica svoltasi a Milano, lo Smau, Salvi non ha saputo contenersi e, arrivato allo stand della Commodore, si è subito misurato con i ragazzini, cercando invano di batterli con un Vic 20. Di fronte ad un Commodore 200, ha perso quasi la ragione, ed il nuovo 64 portatile lo voleva addirittura rubare. ▪

Atterraggio

Gira su Vic inespanso e richiede l'uso dei seguenti comandi: B per iniziare la discesa, S per virare a sinistra e D per girare verso destra.

```
100 REM ATTERRAGGIO
110 REM -----------
120 REM    DI VIZIO
130 REM    MAURIZIO
140 REM TEL 039/743092
141 REM -----------
150 POKE36879,8
160 PRINT"■■
170 IFV=3THEN730
180 FORK=0TO50
190 X%=RND(1)*22
200 Y%=RND(1)*22
210 L=7680+X%+22*Y%
220 POKEL,42:NEXT
230 X=0
240 FORK=XTO21
250 POKE7680+K,160
260 FORC=0TO100:NEXT
270 POKE7680+K,32
280 GETA$
290 IFA$="B"THEN380
300 NEXT
310 FORK=21TOXSTEP-1
320 POKE7680+K,160
330 FORC=0TO100:NEXT
340 POKE7680+K,32
350 GETA$
360 IFA$="B"THEN380
370 NEXT:GOTO240
380 POKE8144,232
390 POKE8145,232
400 POKE8152,232
410 POKE8153,232
420 POKE8160,232
430 POKE8161,232
440 FORY=0TO21
450 L=7680+K+22*Y
460 P=PEEK(L)
470 IFP=42THEN570
480 IFP=232THEN630
490 POKEL,160
500 FORT=0TO300:NEXT
510 POKEL,32
520 GETA$
530 IFA$=""THEN560
540 IFA$="D"THENK=K+1
550 IFA$="S"THENK=K-1
560 NEXT
570 A$="■HAI URTATO"
580 PRINTA$
590 A$="■UNA STELLA"
600 PRINTA$
610 FORT=0TO2500:NEXT
620 V=V+1:GOTO160
630 FORK=0TO100
640 POKE36879,K
650 NEXT
660 POKE36879,81
670 PRINT"■■■■■"
680 A$="ATTERRAGGIO"
690 PRINTA$
700 PRINT"NUMERO";N+1
710 FORT=0TO3000:NEXT
720 N=N+1:GOTO150
730 PRINT"■GIOCO
740 PRINT"TERMINATO
750 PRINT"■PUNTI";N"■
760 FORK=0TO3000:NEXT
770 PRINT"ANCORA";
780 INPUTA$
790 A$=LEFT$(A$,1)
800 IFA$="S"THENRUN
810 POKE36879,27
820 PRINT"■■"
READY.
```

Monitor Tim

Il presente listato avrebbe dovuto essere pubblicato nel fascicolo precedente al quale rinviamo il lettore per comprenderne il funzionamento.

```
100 L0=59:L1=1:E1$="▒                                           ▒":E2$="▒▒ERRORE▒"
110 E3$="INDIR":E4$="L.M.=":E5$="▒ERRORE":E6$=" ▒CONFERMI?":PRINT"▒";
120 PRINTE1$"COMND";:INPUTA1$:L=LEN(A1$):IFL<>4THENPRINTE2$:GOSUB1000:GOTO120
130 IFA1$="MEMO"THENGOSUB1100:GOTO120
135 IFA1$="COMP"THENL0=200:L1=0:GOTO120
140 IFA1$="LARG"THENL0=100:L1=1:PRINT"▒":GOTO120
145 IFA1$="POKE"THENL2=L0:L3=L1:L0=11:L1=1:GOSUB1600:L0=L2:L1=L3:GOTO120
150 IFA1$="RGST"THENGOSUB1800:GOTO120
160 IFA1$="LETT"THENGOSUB2000:GOTO120
990 GOTO120
999 REM *** ATTESA ***
1000 GETX$:IFX$=""THEN1000
1010 RETURN
1020 REM *** VISUALIZZAZIONE MEMORIA ***
1100 PRINTE1$E3$;:INPUTA1$:IFLEN(A1$)>5THENPRINTE2$:GOSUB1000:RETURN
1105 P5=VAL(A1$):FORI=P5TOP5+L0 STEP11:P5=I:GOSUB1110:PRINT"▒"P0$"▒";:FORJ=0TO10
1107 X=I+J:GOSUB1490:PRINT" "P1$;:NEXTJ:IFL1THENPRINT
1109 PRINT:NEXTI:RETURN
1110 P5=P5/256:P6=INT(P5):P7=(P5-P6)*256
1120 P0$="":PE=P6:GOSUB1500:P0$=P0$+P1$
1130 PE=P7:GOSUB1500:P0$=P0$+P1$:RETURN
1490 PE=PEEK(X):REM DE SIMONE SFTWR '83
1500 P1=PE/16:P2=INT(P1):P1=(P1-P2)*16
1510 P1$="":X=P1:GOSUB1550:P1$=PX$
1520 X=P2:GOSUB1550:P1$=PX$+P1$:RETURN
1550 IFX<10THENPX$=CHR$(48+X):RETURN
1560 PX$=CHR$(55+X):RETURN
1570 REM *** OPERAZIONI DI POKE ***
1600 PRINT"▒"E3$;:INPUTA1$:L=LEN(A1$)
1605 IFA1$="←"THENPRINT"▒":RETURN
1610 IFL>5THENPRINTE5$:GOSUB1000:PRINTE5$:GOTO1600
1620 K0=VAL(A1$)
1625 PRINTE3$K0;:INPUTA1$:L=LEN(A1$)
1628 IFA1$="←"THEN1600
1630 IFL<>2THENPRINTE5$:GOSUB1000:GOTO1625
1640 A0=ASC(A1$):A1=ASC(RIGHT$(A1$,1))
1650 IF(A0<48 OR A0>57)AND (A0<65 OR A0>70)THENPRINTE5$:GOSUB1000:GOTO1625
1660 IF(A1<48 OR A1>57)AND (A1<65 OR A1>70)THENPRINTE5$:GOSUB1000:GOTO1625
1670 IFA0<65THENA0=(A0-48)*16:GOTO1690
1680 A0=(A0-55)*16: REM A. DE SIMONE
1690 IFA1<65THENA1=(A1-48):GOTO1710:REM TEL.039/464446
1700 A1=(A1-55):REM LISSONE (MI)
1710 P0=A0+A1:POKEK0,P0:A1$=STR$(K0):K0=K0+1:GOTO1625
1750 REM *** SCRITTURA SU DISCO ***
```

segue a pag. 54 →

A.I.C.A.

computer play® 83

call for games®

SEGRETERIA DEL CONVEGNO

SEGRETERIA AICA
tel. 02/784969-784970
p.le Morandi, 2
20121 Milano

LUOGO DEL CONVEGNO

Palazzo ex-Stelline
c.so Magenta, 61
20123 Milano

L'area del convegno non presenta barriere architettoniche per i portatori di handicap

DATA DEL CONVEGNO

2-3 dicembre 1983

Nell'ambito di COMPUTER PLAY 83 è indetta una gara aperta ai giochi informatici utilizzabili su elaboratori personali (programmi didattici, video games, business games, war games, scacchi, simulazione di gioco a carte, ecc.).
I giochi saranno ammessi previo un controllo da parte del comitato promotore e verranno classificati in tre categorie:

1. Proposti da un singolo appassionato
2. Proposti da gruppi scolastici
3. Proposti da aziende

e verranno valutati in funzione soprattutto dei parametri:

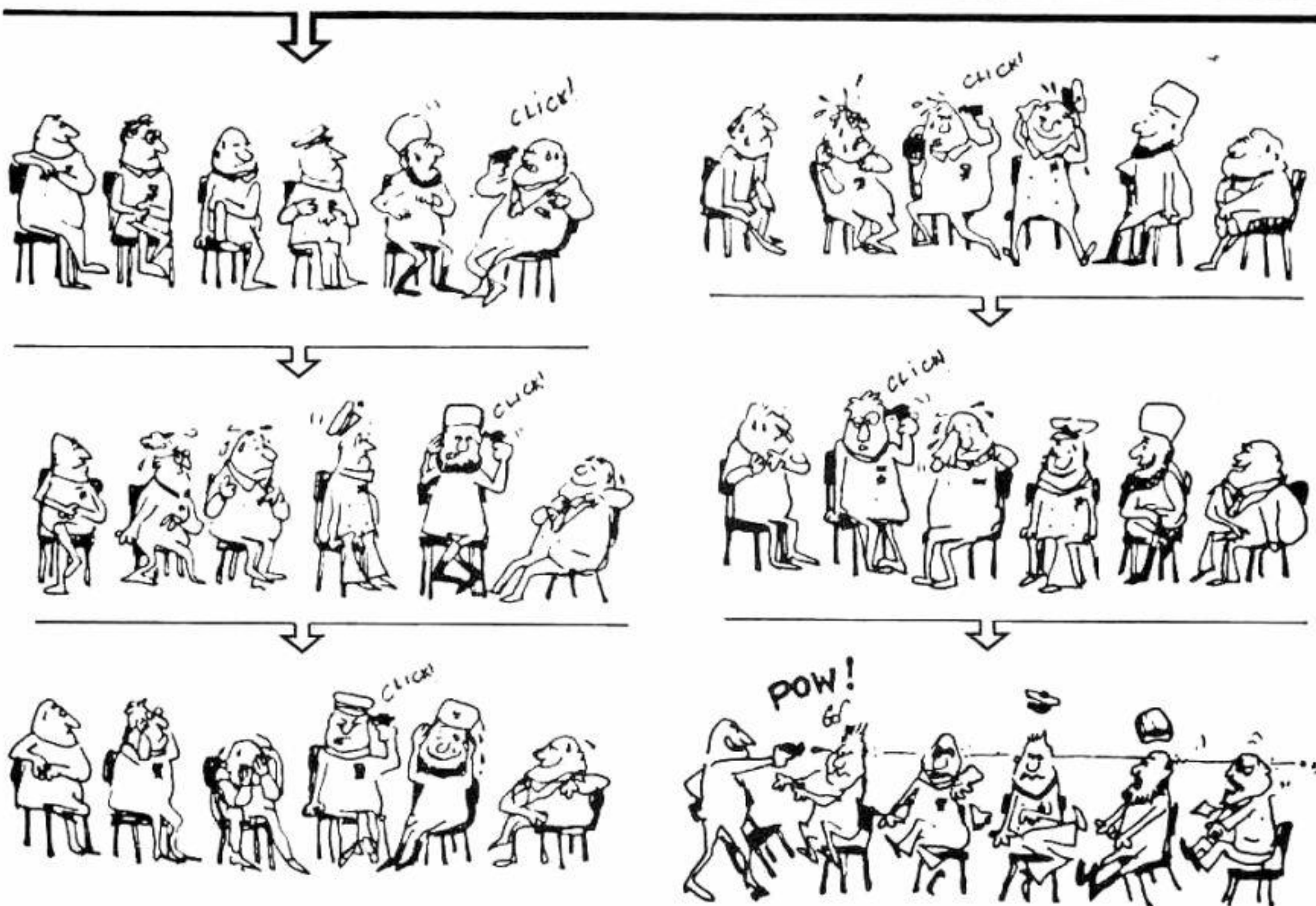
- fascino
- qualità del software
- valore innovativo
- verosimiglianza
- semplicità di impiego

Saranno il pubblico ed il comitato di programma a determinare, con le proprie preferenze, i vincitori nelle varie categorie.

Saranno disponibili in loco i principali elaboratori personali a cui i concorrenti potranno accedere liberamente per provare e dimostrare le loro creazioni.
È indispensabile che il software inerente il gioco proposto sia memorizzato su di un supporto magnetico — cassetta o floppy disk — compatibile con il sistema utilizzato. I concorrenti dovranno segnalare alla segreteria di COMPUTER PLAY 83 la loro intenzione di partecipare alla gara, la categoria in cui intendono concorrere, il nome del gioco proposto ed il sistema su cui è implementato.

LA PARTECIPAZIONE ALLA GARA È GRATUITA.

Roulette russa

E' una versione un po' più sofisticata del noto gioco. Il gioco è sufficientemente descritto dai REMarks presenti nel programma. Va solo precisato che per "sparare" bisognerà usare i tasti da (1) a (6) e non un solo tasto che avrebbe reso il gioco privo di qualsiasi emozione. Quindi caricate questo programma e buona fortuna.

```
15 PRINT "]    R O U L E T T E   R U S S A":PRINT:PRINT:PRINT
20 PRINT "QUESTO PROGRAMMA E' UNA SIMULAZIONE":PRINT
25 PRINT "DELLA NOTA ROULETTE RUSSA.":PRINT:PRINT
30 PRINT "AVETE A DISPOSIZIONE UNA PISTOLA A":PRINT
35 PRINT"SEI COLPI, CARICATA CON UN SOLO":PRINT
40 PRINT "PROIETTILE.":PRINT:PRINT
45 PRINT "PER SPARARE, ATTENDETE IL VOSTRO TURNO,":PRINT
50 PRINT "E PREMETE I TASTI DA ['1'] A ['6'].
55 PRINT:PRINT:PRINT"*******   BUONA  FORTUNA   *******"
```

```
57 FOR I=1 TO 5000:NEXT I:GOSUB 1400
60 REM ---------------------------------
61 REM -         INPUT  DATI          -
62 REM ---------------------------------
65 PRINT "□":INPUT"NOME DEL 1° GIOCATORE ";G1$:G1$="▣  "+LEFT$(G1$,8)+"  ":PRINT
70 INPUT "NOME DEL 2° GIOCATORE ";G2$:G2$="▣  "+LEFT$(G2$,8)+", ":PRINT
75 INPUT "PARTITE DA EFFETTUARE ";N:IF N/2-INT(N/2)<>0 THEN 77
76 N=N+1
77 PRINT "□"
80 REM ---------------------------------
81 REM -   STABILISCE CHI INCOMINCIA  -
82 REM ---------------------------------
85 I%=RND(1)*2+1:IF I%=1 THEN T=2:GOTO 116
90 T=1
105 REM ---------------------------------
106 REM -        INIZIO  GIOCO         -
107 REM ---------------------------------
110 CO%=RND(1)*6+1:REM * RUOTA IL TAMBURO *
120 T=T+1:IF T>2 THEN T=1
140 IF T=1 THEN D$="◣":GOSUB 480:PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"TAB(15)G1$:GOTO 170
150 C$="◣":D$="":GOSUB480:PRINT"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"TAB(15)G2$:C$="":D$="◣"
170 PRINT "▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒" TAB(6) "****  PREMI IL GRILLETTO  ****"
180 GET A$:IF A$="" THEN 180
190 IF ASC(A$)<49 OR ASC(A$)>54 THEN 180
195 PRINT "▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒" TAB(15) "                              "
200 C%=VAL(A$)
210 IF C%=CO% THEN GOTO 900
220 GOSUB 800
230 GOTO 110
240 REM ---------------------------------
241 REM -   AGGIORNAMENTO RISULTATI   -
242 REM ---------------------------------
250 IF T=1 THEN G2=G2+1:GOTO 270
260 G1=G1+1
270 IF G1>INT(N/2) OR G2>INT(N/2) THEN 300
280 GOTO 110
290 REM ---------------------------------
291 REM -          FINE PARTITA        -
292 REM ---------------------------------
300 IF G1>G2 THEN A$=G1$:B$=G2$:GOTO 320
310 A$=G2$:B$=G1$
320 PRINT "□" TAB(15) "V I N C E":PRINT:PRINT
322 E$="O":IF LEFT$(RIGHT$(A$,3),1)="A" THEN E$="A"
325 PRINT TAB(14) A$:PRINT:PRINT
330 PRINT TAB(4) "PER ESSER MORT" E$ " MENO VOLTE DI":PRINT:PRINT
340 PRINT TAB(14) B$ :PRINT:PRINT:PRINT:PRINT
350 PRINT"*** VOLETE GIOCARE ANCORA  ? [ S-N ] ***"
360 GET A$:IF A$="" THEN 360
370 IF ASC(A$)=83 THEN PRINT "□":G1=0:G2=0:GOTO 85
375 IF ASC(A$)=78 THEN PRINT TAB(9) "▒▒*****  C I A O  *****"
377 FOR I=1 TO 1000:NEXT:PRINT "□":END
```

⇨

```
380 GOTO 360
450 REM ------------------------------
455 REM -     STAMPA TESTA E PISTOLA    -
460 REM ------------------------------
480 PRINT "";
490 PRINT TAB(8) C$ "_______"
500 PRINT TAB(7) C$ "/XXXXXXX\"
510 PRINT TAB(6) C$ "/XXXXXXXXX\"
520 PRINT TAB(6) D$ "           "
530 PRINT TAB(5) C$ " X""|          |" C$ "X "
540 PRINT TAB(5) " X""|  O     O  |" C$ "X "
550 PRINT TAB(5) " X""|          |" C$"X  " ""
560 PRINT TAB(6) C$ "¯""|    /\    |" C$ "¯" ""
570 PRINT TAB(7) "|          |"
580 PRINT TAB(7) "|   ____   |"
590 PRINT TAB(7) "\  \__/  /"
600 PRINT TAB(8) "\      /"
610 PRINT TAB(9) "\____/"
620 PRINT TAB(11) "| | "
700 PRINT "";
710 PRINT TAB(25) "         \"
720 PRINT TAB(28) "/\ |"
730 PRINT TAB(30) "| |"
740 PRINT TAB(30) "|_|"
755 PRINT "" G1$ G1 TAB(20) G2$ G2
760 RETURN
780 REM ------------------------------
781 REM -        COLPO A VUOTO         -
782 REM ------------------------------
800 PRINT "":POKE V1,1:POKE W,129:POKE A,128:POKEH,217:POKEL,189
820 POKE V1,0:POKE W,0:POKEA,0:RETURN
895 REM ------------------------------
896 REM -           SPARO              -
897 REM ------------------------------
900 PRINT ""
910 FOR S=15 TO 0 STEP -1:POKE V1,S:POKE W,129:POKE A,15:POKE H,40:POKE L,200
920 NEXT S:POKE W,0:POKE A,0
1000 POKE V+41,14
1010 IF T=2 THEN POKE V+41,3
1020 POKE V+21,4:POKE V+5,98:POKE V+27,4
1030 FOR X=200 TO 10 STEP -2:POKE V+4,X:NEXT X:POKE V+21,0
1040 GOTO 250
1390 REM ------------------------------
1392 REM -   COSTRUZIONE  PROIETTILE  -
1395 REM ------------------------------
1400 V=53248:POKE 2042,13
1410 FOR N=0 TO 23:READ Q:POKE 832+N,Q:NEXT N
1500 DATA 7,255,255,31,255,255,127,255,255,255,255,255
1510 DATA 255,255,255,127,255,255,31,255,255,7,255,255
1600 V1=54296:W=54276:A=54277:H=54273:L=54272:RETURN

READY.
```

Gerardo Marra - Via P. Baratta, 91/A - Battipaglia

Il sistema solare visibile

Un videogioco istruttivo per Vic 20 e Commodore 64.

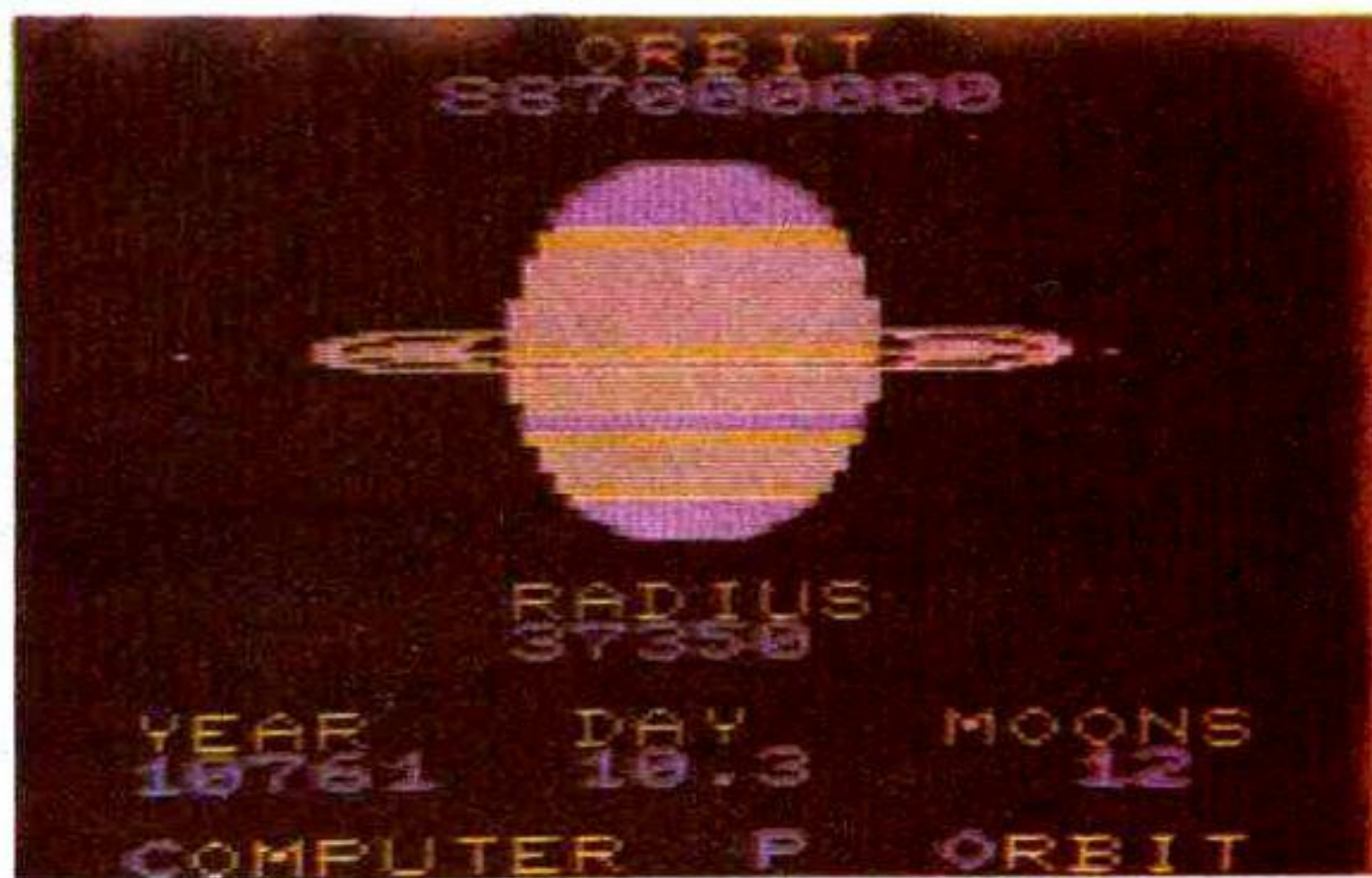
E' UNO dei più eccitanti fra i nuovi giochi introdotti dalla Commodore per il 64 ed il Vic 20, ma proprio per questo è anche uno dei più difficili. Richiede tutte le abilità dei videogiochi più complessi, ma al contempo vi obbliga ad imparare e a conoscere il sistema solare come un astronomo o un astronauta.

Tu sei il comandante d'una navicella spaziale in viaggio attraverso il sistema solare. L'astronave ha una velocità di crociera d'oltre un miliardo di miglia ed è equipaggiata con un sistema computerizzato, che vi consente sempre nuove ed eccitanti scoperte.

Quando viene inserita la cartuccia, visualizza Mercurio, Venere, la Terra e Marte, che ruotano attorno al sole. Subito dopo appare la vostra astronave con, in primo piano, il pianeta Terra ed una serie di dati circa la sua dimensione ed il suo moto rotatorio. Nel corso del viaggio potrete visitare gli altri pianeti del sistema solare.

Per cominciare a giocare dovete premere F1.

Navigazione

Per navigare disponete d'uno schermo di controllo che consente di controllare e modificare la

posizione della navicella, muovendovi dal centro dello schermo verso le orbite di Mercurio, Venere, Terra, Marte e Giove. I numeri, che compaiono accanto ad ogni pianeta, indicano la distanza dal sole delle rispettive orbite in milioni di miglia. In rosso è segnata la posizione della vostra navicella. Sulla sinistra, in alto, è segnata l'altezza dell'astronave (sempre in milioni di miglia) rispetto all'orbita dei pianeti.

Una telecamera rotante, installata sulla navicella, consente d'osservare le orbite ed i movimenti dei pianeti in una prospettiva tridimensionale. A sinistra, in basso, un apposito numerino blu indica l'angolo verso cui è rivolta la telecamera: O significa che sta guardando perpendicolarmente sotto l'astronave: man mano che raggiunge gli 80 gradi, l'obiettivo si sposta fino a puntare ciò che sta davanti. Per vedere le orbite dei pianeti, occorre premere il tasto V (da view = vista). Per tornare alla strumentazione bisogna battere O.

I comandi per azionare la navicella sono:

CRSR → per orientare la navicella verso destra (l'orientamento è segnato dal punto rosso):

CRSR ↓ per invertire la direzione:

→ per virare a sinistra:

↓ per andare avanti:

G ordina alla navicella di spostarsi nella direzione segnata prima:

U muove la navicella verso l'alto:

D muove la navicella verso il basso.

REGALATI CO

Quest'anno, fatti un regalo intelligente: un computer dalle caratteristiche incredibili. Vediamole.

1. Commodore 64 è potente, sofisticato, professionale.

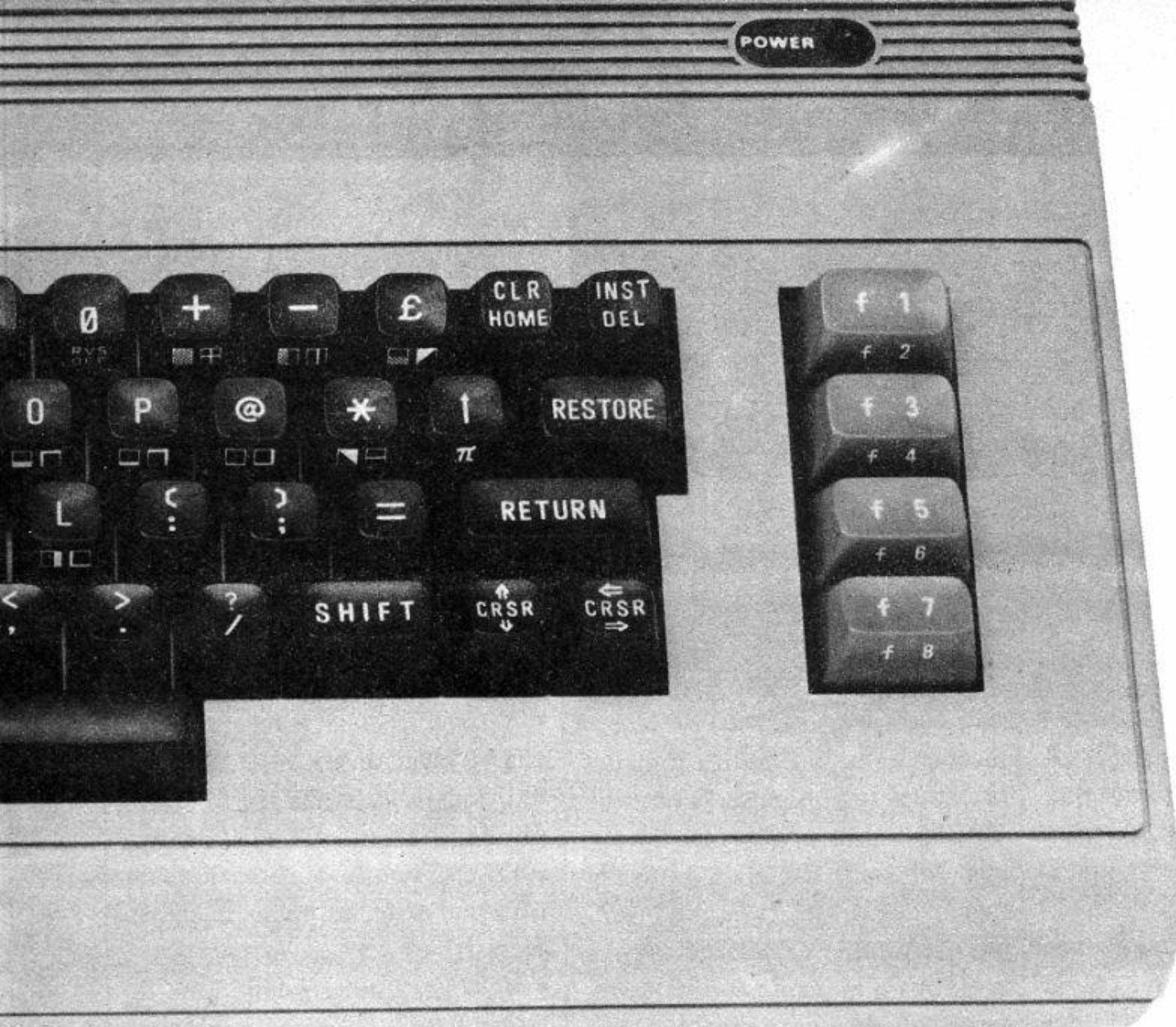
2. Ha una vastissima gamma di programmi già pronti, lo usi nella professione, a casa, a scuola, nella ricerca scientifica, con facilità e totale affidabilità.

3. Ha un'incredibile memoria (64 K), un sintetizzatore sonoro professionale, produce effetti tridimensionali.

4. Ti diverti perchè è anche un sofisticato videogioco.

5. Con Commodore 64 entri nel futuro, tasto dopo tasto.

6. Oggi Commodore 64 ha un prezzo incredibile: per L. 625.000 + IVA ti trovi anche un registratore, un videogioco e il 1° volume "introduzione al BASIC".

Quasi 230.000 lire risparmiate.
Non è incredibile?

MMODORE 64.

L'incredibile.

L. 625.000 più IVA.
Compresi nel prezzo, registratore, videogioco e un volume del BASIC.

commodore
COMPUTER

ta, o L (less = meno), se va diminuita.

Il computer planetario

Se vuoi maggiori informazioni sui sei principali pianeti (Mercurio, Venere, Terra, Marte, Giove e Saturno) o vuoi fare dei raffronti tra gli stessi, batti un numero da 1 a 6. I dati visualizzati sono:

- *RADIUS*, ossia il raggio in miglia. I raggi di Giove e Saturno, indicano la distanza tra lo strato più denso di nuvole ed il centro;
- *YEAR*, tempo di roteazione attorno al sole, espresso in giorni terrestri;
- *MOONS*, numero delle lune relative;
- *ORBIT*, velocità di roteazione del pianeta lungo la propria orbita in miglia per secondo;
- *ESCAPE*, velocità (in miglia per secondo) necessaria ad un'astronave per sfuggire alla forza di gravità del pianeta;
- *TO ATM*, temperatura in gradi Farenheit sulla superficie delle nuvole del pianeta;
- *WARM ATM*, temperatura massima in Farenheit;
- *SURFACE*, temperatura sulla superficie del pianeta.

Se invece di numeri compare un asterisco, significa che quella precisa informazione non è ancora disponibile.

Per mettere a confronto i dati di un pianeta già selezionato con quelli di un altro, occorre battere A, finchè questo pianeta non compare sotto la scritta ASTROCALC. ■

La navicella è in grado di effettuare due itinerari animati preprogrammati. In tal caso, il tasto S consente di esplorare la superficie del sistema a distanza ravvicinata. Quando il giro è finito premete O. Col comando A, la navicella si allontanerà di 500 milioni di miglia dall'orbita di Giove, verso l'orbita della Terra. Quando lo spostamento è finito, occorre battere ancora O. Questi spostamenti lasciano l'astronave a soli 15 milioni di miglia dalle orbite solari, cioè molto più in basso di dove potete arrivare pilotando voi stessi l'astronave. Volendo, potete effettuare le vostre esplorazioni da questa altezza, oppure, potete ritornare a 30 milioni di miglia battendo F3 ed F1.

I pianeti in primo piano

Per vedere da vicino i pianeti Terra, Giove e Saturno, è sufficiente battere P. Lo stesso tasto serve per cambiare pianeta.

Sullo schermo in alto, si vede il diagramma del sole (a sinistra) e dei sette pianeti più vicini. Una piccola stella blu, indica il pianeta che avete zoomato. Il termine ORBIT indica la distanza in miglia dal sole; RADIUS indica il raggio dell'orbita (sempre in miglia). L'anno (YEAR) ed il giorno (DAY), indicano la data terrestre. Solo alcune lune (MOONS) di Giove e Saturno, possono essere viste dall'astronave.

Ogni pianeta ha una propria forza di gravità, e WEIGHT (peso), indica quanto pesereste su quel pianeta, supposto che, sulla terra pesiate 160 libre (ogni libra è pari a 453 grammi). AGE, indica la vostra età in anni di quel pianeta, supponendo che sulla terra ne abbiate 24.

A tale scopo, prima di visualizzare i vari dati, il computer planetario vi chiede di segnalare quale pianeta vi interessa (battendo il tasto P di PLANET), e di fornire i vostri dati personali (battere Y per richiederli) ed in particolare l'età (Age) ed il peso (Weight). Per fornire questi ultimi dati dovete digitare C (da correct, esatto) se la cifra proposta dal computer è giusta, oppure M (more = di più) se va accresciu-

Antiaerea

```
10 REM - ANTIAEREA PER VIC 20  NON ESPANSO -
12 REM BALDI EMANUELE  . TEL 039/749652
13 :
14 REM F1=PER PUNTARE IL CANNONE IN DIAGONALE
18 REM F3=PER PUNTARE IL CANNONE IN AVANTI
22 REM F5=PER PUNTARE IL CANNONE VERSO L' ALTO
1000 REM   SPAZIO PER
1001 REM   SPARARE .
1002 Z=36878:O=36876:B=36877
1010 POKE36879,14:PRINT"□":PRINT"■■■■      ANTIAEREA":GOSUB1420
1020 PRINT"■DIFFICOLTA'(1-100)":INPUTD
1030 IFD<20THENS=20:GOTO1050
1040 S=D
1050 PRINT"HAI";S;"COLPI"
1060 PRINT"■   PREMI UN TASTO"
1070 GETB$:IFB$=""THENGOTO1070
1080 PRINT"□":POKE36869,254:POKE56,24:POKE55,0:POKE650,255:Q=8186:H=35
1090 FORK=0TO23:READA:POKE6144+8*32+K,A:NEXTK
1100 DATA 0,0,0,0,0,0,0,0,24,24,24,24,24,24,0,24,16,12,12,255,64,35,20,8
1110 FORK=0TO23:READA:POKE6144+8*35+K,A:NEXTK
1120 DATA0,60,12,127,64,35,52,104,56,95,127,255,128,89,20,0,62,62,20,255,1,2,2,1
64
1130 FORK=0TO23:READA:POKE6144+8*38+K,A:NEXTK
1140 DATA64,224,112,56,28,8,1,2,0,0,231,24,255,24,36,36,56,110,192,170,213,192,1
10,56
1150 FORK=0TO15:READA:POKE6144+K+8*41,A:NEXTK:DATA4,4,12,127,64,35,52,104
1160 DATA0,253,253,0,0,0,0,0
1170 POKE7890,39
1180 FORK=0TO506:IFPEEK(Q-K-1)<>32THEN1280
1190 POKEQ-K,H:POKEQ+1-K,36:POKEQ+2-K,37
1200 GETA$
1210 FORF=0TOD:NEXTF
1220 IFA$=" "THENGOSUB1290
1230 IFA$="■"THENH=34
1240 IFA$="■"THENH=35
1250 IFA$="■"THENH=41
1260 POKEQ-K,32:POKEQ+1-K,32:POKEQ+2-K,32
1270 IFNOTV>STHENNEXTK
1280 PRINT"□":POKE36869,240:PRINT"HAI DISTRUTTO";P:PRINT"AVVERSARI":END
1290 IFH=34THENGOSUB1390:FORX=1TO23:IFPEEK(Q-K-X*23)=39THEN1370
1300 IFH=34THENPOKEQ-K-X*23,38:POKEQ-K-X*23,32:NEXTX:V=V+1:IFP>3ANDC<70THENPOKEQ
-K-16,40
1310 IFH=34THENRETURN
1320 IFH=41THENGOSUB1390:FORX=1TO22:IFPEEK(Q-K-22*X)=39THEN1380
1330 IFH=41THENPOKEQ-K-X*22,33:POKEQ-K-X*22,32:NEXTX:V=V+1:IFP>3ANDC<70THENPOKEQ
-K-16,40
1340 IFH=41THENRETURN
1350 GOSUB1390:FORX=1TO10:IFPEEK(Q-K-X)=40THENP=P+1:POKEQ-K-X,32:GOSUB1410
1360 POKEQ-K-X,42:FORR=0TO5:NEXTR:POKEQ-K-X,32:NEXTX:V=V+1:RETURN
1370 P=P+1:POKEQ-K-23*X,32:C=RND(1)*110:POKE7680+C,39:GOSUB1410:RETURN
1380 P=P+1:POKEQ-K-22*X,32:C=RND(1)*110:POKE7680+C,39:GOSUB1410:RETURN
1390 POKEZ,15:FORL=1TO30:FORM=250TO240STEP-1:POKEO,M:NEXTM:FORM=240TO250
1400 POKEO,M:NEXTM:POKEZ,0:RETURN
1410 POKEO,0:POKEB,220:FORL=10TO0STEP-1:POKEZ,L:FORM=1TO130:NEXTM,L:POKEB,0:RETU
RN
1420 FORG=0TO3:POKEZ,15:FORL=1TO10:FORM=180TO235STEP2:POKEO,M:FORN=1TO10:NEXTN,M
:POKEO,0
1430 FORM=1TO100:NEXTM:POKEZ,0:NEXTG:RETURN
```

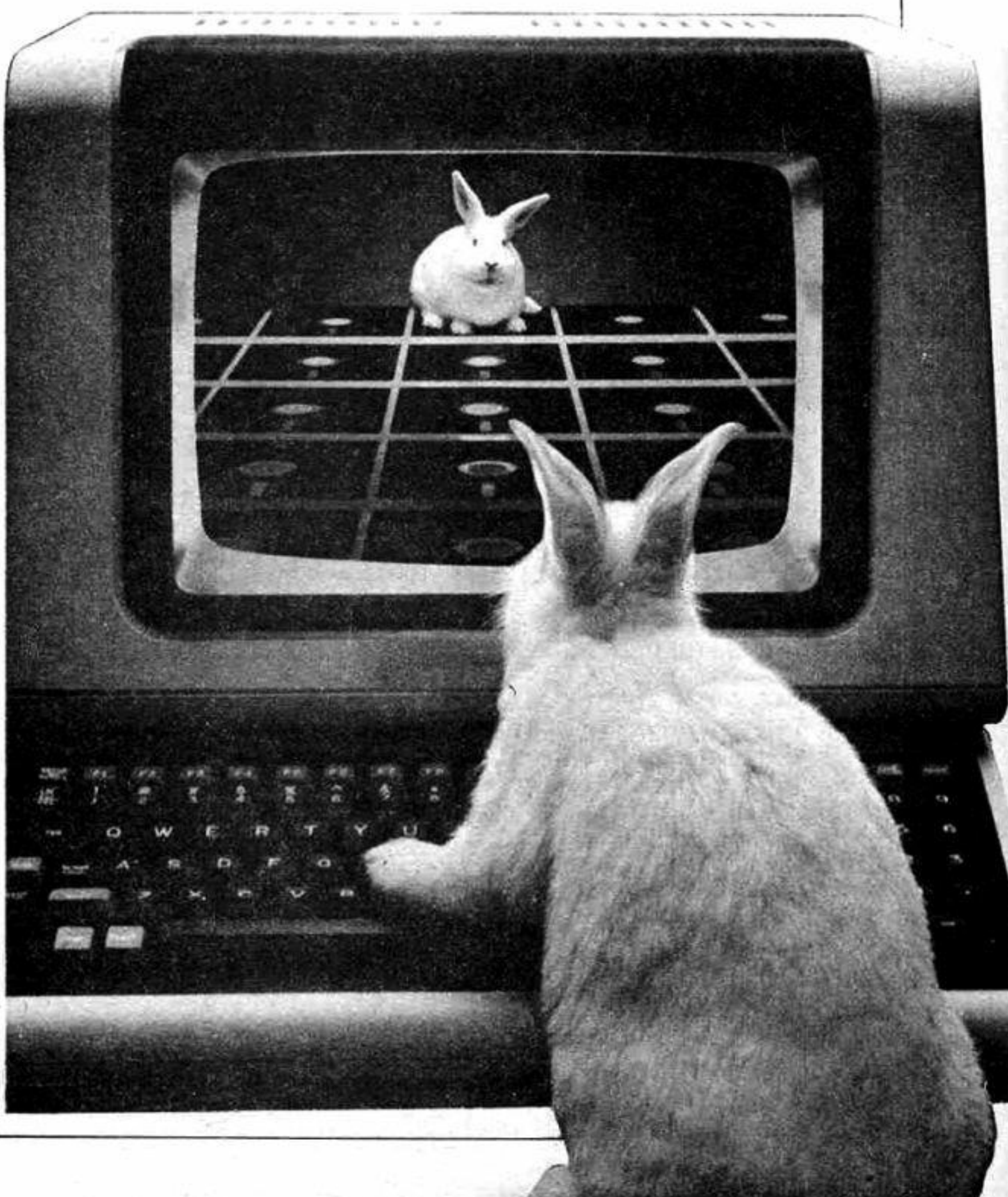
Forza 4

Un programma che è stato fatto per il PET, e di cui viene data anche la versione per il fratello minore VIC 20.

IL GIOCO consiste nel riuscire a mettere 4 pedine dello stesso giocatore di seguito, sia orizzontalmente, verticalmente, che diagonalmente.

Detto così può apparire facile: sembra un'ennesima edizione del tris, solo che invece di 3, sono 4 le pedine da mettere in fila. Ma non è affatto così: infatti, sul video ci sono 12 colonne (10 per ovvi motivi nella versione per il VIC), e ad ognuna di esse corrisponde una lettera dell'alfabeto (dalla A alla L), in cui si lasciano cadere le pedine, una alla volta a turno per i due giocatori.

Queste pedine non si fermano dove vogliono, nè dove vorremmo si fermassero, ma hanno un ordine ben preciso: la prima, che per esempio metteremo nella colonna B, andrà sul fondo; la seconda (sempre in B), andrà immediatamente sopra alla prima, e così via.

Può capitare a volte di non accorgersi di aver vinto - e soprattutto di avere perso - perchè si era troppo attenti a seguire il

proprio piano di gioco, ed è il computer che ce lo fa capire con lampeggi delle 4 pedine in fila e con suoni.

All'inizio bisogna dare il nome dei due giocatori, quindi, inizia subito la partita: verranno disegnate le colonne sul video e la scritta "player 1" apparirà in basso a destra. Bisogna quindi dire al computer in quale colonna vogliamo mettere la nostra pedina - premendo il tasto relativo alla lettera indicante la colonna - ed allora la si vedrà cadere dall'alto della colonna scelta. Poi comparirà la scritta "player 2" e toccherà quindi al secondo giocatore, e così via fino a quando uno dei due non avrà vinto.

Quando una colonna è piena non vi si possono più mettere pedine e, se si tenta , il computer ci dice di rifare. Alla fine del gioco viene data la classifica delle vittorie dei giocatori e viene chiesto se si vuole giocare ancora e, nel caso affermativo, si comincierà una nuova partita.

Come funziona

Dalla linea 150 alla 220, vengono chiesti i nomi dei due giocatori (G1$ e G2$), e vengono formate le stringhe A$ e D$, che servono per i print sul video.

Da 250 a 300 vengono visualizzate le colonne.

Da 310 a 450 viene chiesto il nome della colonna in cui si vuol mettere la pedina, la quale viene fatta cadere (390-450).

Le linee da 470 a 740 servono per controllare se un giocatore ha vinto, cioè se ha messo 4 pedine di seguito.

Il procedimento per controllare questo è abbastanza semplice: supponiamo che il computer stia controllando se le 4 pedine sono in fila orizzontalmente, (verticalmente e diagonalmente, il procedimento è il medesimo).

Partendo dalla pedina che sta immediatamente di fianco a quella appena introdotta, prosegue fino a quando ne trova una dell'avversario o fino a quando non c'è alcuna pedina, e poi torna indietro contando le pedine: se risultano essere 4 (o anche di più, come può capitare), vuol dire che si è vinto.

La linea 990 si è resa necessaria per il caso in cui si voglia mettere la pedina in una colonna già piena.

Versione Vic

Le uniche varianti riguardano i valori dei POKE sia per il video che per il suono. Per il resto la versione per il Vic è essenzialmente identica a quella del PET. ▪

```
10 POKE36879,8 :PRINT"■"
100 REM ************************
110 REM ** (C) BELLU' GIOVANNI **
115 REM **   VERSIONE  VIC 20   **
116 REM **     RIVEDUTA DA      **
117 REM **    CIANI GIUSEPPE    **
120 REM ************************
130 PRINT"■"
140 FORK=1TO20:D$=D$+" ":NEXT
150 PRINT"■NOME PRIMO GIOCATORE■■":INPUTG1$
170 PRINT"■■■NOME SECONDO GIOCATORE■■":INPUTG2$
180 POKE36878,15
190 SS=36876
200 PRINT"■"
210 FORK=1TO15:PRINT"| | | | | | | | | | |":NEXT
220 PRINT"└┴┴┴┴┴┴┴┴┴┘"
230 A$="■":FORK=1TO17:A$=A$+"■":NEXT
240 B$="●":C$="O":A=7724:B=81:C=87
250 PRINTA$D$
260 PRINT"■■ A B C D E F G H I J"
```

➔

```
270 U=1:PRINTA$"PLAYER 1       ▓▓▓▓▓▓";:GOSUB310
280 D=B:GOSUB330
290 D=C:U=2:PRINTA$"PLAYER 2        ▓▓▓▓▓▓";:GOSUB310:GOSUB330:GOTO270
300 END
310 GETM$:IFM$<"A"ORM$>"J"THEN310
320 PRINTM$;:RETURN
330 REM
340 W=A+1+(ASC(M$)-65)*2
350 FORK=WTO8073STEP22:I=I+1
360 IFPEEK(K)<>32THENPOKEK-22,D:W=K-22:K=8073:GOTO0380
370 POKEK,D:FORPP=1TO10:NEXT
380 POKESS,135+I*4:FORO=1TO2*I:NEXT:POKESS,0:POKEK,32:NEXT:I=I+2
390 POKESS,135+I*4:FORJ=1TO20*I:NEXT:POKESS,0:IFI<4THENPRINT"RIFAI":I=0:GOTO1000
395 I=0
400 FORJ=1TO4
410 IFPEEK(W+J*2)<>DTHENA1=J:J=4:GOTO430
420 NEXTJ:J=0:GOTO440
430 NEXTJ:J=0
440 FORJ=1TO4
450 IFPEEK(W+(A1)*2-J*2)=DTHENR=R+1:IFR=4THENPRINT"▓HAI VINTO":P=2:GOTO680
460 NEXT:R=0
470 J=0
480 FORJ=1TO4
490 IFPEEK(W+20*J)<>DTHENA2=J:J=4
500 NEXTJ:J=0
510 FORJ=1TO4
520 IFPEEK(W+A2*20-J*20)=DTHENR=R+1:IFR=4THENPRINT"▓HAI VINTO":P=20:GOTO680
530 NEXT:R=0
540 J=0
550 FORJ=1TO4
560 IFPEEK(W+22*J)<>DTHENA3=J:J=4
570 NEXTJ:J=0
580 FORJ=1TO4
590 IFPEEK(W+A3*22-J*22)=DTHENR=R+1:IFR=4THENPRINT"▓HAI VINTO":P=22:GOTO680
600 NEXT:R=0
610 FORJ=1TO4
620 IFPEEK(W+24*J)<>DTHENA4=J:J=4
630 NEXTJ:J=0
640 FORJ=1TO4
650 IFPEEK(W+A4*24-J*24)=DTHENR=R+1:IFR=4THENPRINT"▓HAI VINTO":P=24:GOTO680
660 NEXT:R=0
670 RETURN
680 IFP=2THENX=A1
690 IFP=20THENX=A2
700 IFP=22THENX=A3
710 IFP=24THENX=A4
720 J=4:NEXT
730 FORZ=1TO30
740 FORJ=1TO4
750 POKEW+X*P-J*P,42
760 POKESS,75:FORH=1TO1:NEXT:POKESS,0:NEXTJ
770 FORK=1TO10:NEXT
```

```
780 FORJ=1TO4
790 POKEW+X*P-J*P,D
800 POKESS,174:FORH=1TO1:NEXT:POKESS,0:NEXTJ
810 NEXTZ
820 IFD=81THENP1=P1+1:GOTO840
830 P2=P2+1
840 PRINT"□"
850 PRINT"PLAYER 1   :":PRINTG1$:PRINT"PUNTI  ";P1
860 PRINT"PLAYER 2   :":PRINTG2$:PRINT"PUNTI  ";P2
870 PRINT"ANCORA ? (S/N)"
880 GETA$:IFA$=""THEN880
890 IFA$<>"S"ANDA$<>"N"THEN880
900 IFA$="N"THENEND:END:END
910 GOTO180
1000 POKEW,32: FORQ=1TO1000:NEXT:ONUGOTO270,290

READY.
```

VERSIONE PET

```
        *************************
150 PRINT"□"
160 A$="":FORK=1TO17:A$=A$+"":NEXT
170 PRINT""TAB(10)"F  O  R  Z  A  -  4  "
180 PRINT:PRINTTAB(8)"BY BELLU' GIOVANNI"
190 FORK=1TO40:D$=D$+" ":NEXT
200 PRINTA$"NOME PRIMO GIOCATORE";:INPUTG1$
210 PRINTA$D$
220 PRINTA$"NOME SECONDO GIOCATORE";:INPUTG2$
230 POKE59467,16:POKE59466,16
240 SS=59464
250 PRINT"□"
260 FORK=1TO15:PRINT"| | | | | | | | | | | | |":NEXT
270 PRINT"└┴┴┴┴┴┴┴┴┴┴┴┘"
280 B$="●":C$="o":A=32848:B=81:C=87
290 PRINTA$D$
300 PRINT" A B C D E F G H I J K L"
310 U=1:PRINTA$"PLAYER 1       ";:GOSUB350
320 D=B:GOSUB370
330 D=C:U=2:PRINTA$"PLAYER 2       ";:GOSUB350:GOSUB370:GOTO310
340 END
350 GETM$:IFM$<"A"ORM$>"L"THEN350
360 PRINTM$:RETURN
370 REM
380 W=A+1+(ASC(M$)-65)*2
390 FORK=WTO33500STEP40:I=I+1
400 IFPEEK(K)<>32THENPOKEK-40,D:W=K-40:K=33500:GOTO420
410 POKEK,D:FORPP=1TO10:NEXT
420 POKESS,15+I*10:FORO=1TO2*I:NEXT:POKESS,0:POKEK,32:NEXT:I=I+2
430 POKESS,15+I*10:FORJ=1TO20*I:NEXT:POKESS,0
440 IFI=4THENPOKEW-40,160
450 IFI<4THENPRINTA$"RIFAI                    ":I=0:GOTO990
```

```
 460 I=0
 470 FORJ=1TO4
 480 IFPEEK(W+J*2)<>DTHENA1=J:J=4:GOTO500
 490 NEXTJ:J=0:GOTO510
 500 NEXTJ:J=0
 510 FORJ=1TO4
 520 IFPEEK(W+(A1)*2-J*2)=DTHENR=R+1:IFR=4THENPRINT"HAI VINTO":P=2:GOTO750
 530 NEXT:R=0
 540 J=0
 550 FORJ=1TO4
 560 IFPEEK(W+38*J)<>DTHENA2=J:J=4
 570 NEXTJ:J=0
 580 FORJ=1TO4
 590 IFPEEK(W+A2*38-J*38)=DTHENR=R+1:IFR=4THENPRINT"HAI VINTO":P=38:GOTO750
 600 NEXT:R=0
 610 J=0
 620 FORJ=1TO4
 630 IFPEEK(W+40*J)<>DTHENA3=J:J=4
 640 NEXTJ:J=0
 650 FORJ=1TO4
 660 IFPEEK(W+A3*40-J*40)=DTHENR=R+1:IFR=4THENPRINT"HAI VINTO":P=40:GOTO750
 670 NEXT:R=0
 680  ORJ=1TO4
 690 IFPEEK(W+42*J)<>DTHENA4=J:J=4
 700 NEXTJ:J=0
 710 FORJ=1TO4
 720 IFPEEK(W+A4*42-J*42)=DTHENR=R+1:IFR=4THENPRINT"HAI VINTO":P=42:GOTO750
 730 NEXT:R=0
 740 RETURN
 750 IFP=2THENX=A1
 760 IFP=38THENX=A2
 770 IFP=40THENX=A3
 780 IFP=42THENX=A4
 790 J=4:NEXT
 800 FORZ=1TO30
 810 FORJ=1TO4
 820 POKEW+X*P-J*P,42
 830 POKESS,75:FORH=1TO1:NEXT:POKESS,0:NEXTJ
 840 FORK=1TO10:NEXT
 850 FORJ=1TO4
 860 POKEW+X*P-J*P,D
 870 POKESS,174:FORH=1TO1:NEXT:POKESS,0:NEXTJ
 880 NEXTZ
 890 IFD=81THENP1=P1+1:GOTO910
 900 P2=P2+1
 910 PRINT""
 920 PRINT"PLAYER 1   :";G1$;" = PUNTI  ";P1
 930 PRINT"PLAYER 2   :";G2$;" = PUNTI  ";P2
 940 PRINT"ANCORA ?"
 950 GETQ$:IFQ$=""THEN950
 960 IFQ$<>"S"ANDQ$<>"N"THEN950
 970 IFQ$="N"THEMEND:END:END
 980 GOTO230
 990 POKEW,160:FORQ=1TO1000:NEXT:ONUGOTO290,310
READY.
```

IMPARA A PROGRAMMARE CON IL VIC

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

★ DISPENSA N. 6 ★

Se stai comprando un personal computer prova a farti queste domande:
1. Chi è oggi il più affidabile?
2. Chi dà la possibilità di scegliere fra più sistemi?
3. Chi fornisce soluzioni, subito, in una gamma vastissima?
4. Chi propone il miglior rapporto fra costi e prestazioni?
5. Chi ti dà una così grande esperienza ed assistenza?
A tutte le domande puoi rispondere con

ORE, IL N°1

un solo nome: Commodore Computer.
Anche per questo Commodore
è il Numero 1. In Europa e in Italia.
Sei in buone mani.
Commodore Italiana Spa
Milano, telefono 02/6125651

7.Le altre funzioni (seconda parte)

Funzioni definite dall'utente

Il Computer ha la possibilità di utilizzare funzioni definite dall'utente mediante l'uso dell'istruzione DEF FNx, che assume la forma generale seguente:

numero di linea DEF FNx (variabile) = espressione

Tutte le funzioni definite in questo modo devono avere all'interno del programma un solo nome definito da FNx, dove x e il nome di variabile.

Tutte le funzioni hanno un argomento fittizio dato dalla variabile racchiusa fra parentesi, mentre quello reale, usato quando la funzione viene successivamente menzionata nel programma, sarà differente dal primo nella definizione della funzione. Per esempio, l'espressione precedente, usata per arrotondare un numero, può essere definita come funzione con l'istruzione seguente:

50 DEF FNR(A)=INT(A/F + 0,5)*F

che può venire in seguito utilizzata nello stesso programma per arrotondare un numero, ad esempio alle centinaia e ad una cifra decimale come riportato nella tabella 7.5.

```
140 F=0.1
150 C1=FNR(C)
160 PRINT "B=";B" B1=";B1
165 PRINT "C=";C;" C1=";C1
170 PRINT
180 GOTO 60
190 END
```

Tabella 7.5 Programma di arrotondamento di numeri

Le funzioni definite dall'utente possono contenere a loro volta, nella loro definizione, altre funzioni dello stesso tipo. Per esempio, il programma della tabella 7.6 tabula i seni di un certo numero di angoli (noti) e gli angoli corrispondenti espressi in gradi alla seconda cifra decimale.

```
20 PRINT"   SENO  GRADI"
30 PRINT"   ----  -----"
40 PRINT
50 DEF FNR(A)=INT(A/0.1+0.5)*0.1
60 DEF FND(U)=180/(ATN(1)*4)
70 DEF FNA(S)=(ATN(S/SQR(1-S*S)))*FND(U)
80 INPUT X
90 IF X=0 THEN 160
100 IF X<1 THEN 130
105 IF X>1 THEN PRINT"ERRORE": GOTO 80
110 D=90
120 GOTO 140
130 D=FNR(FNA(X))
140 PRINT TAB(1);X;TAB(9);D
150 GOTO 80
160 END
```

Tabella 7.6 Seni e angoli (in gradi)

Esercizi

Problema 1 - Raggio di un cerchio circoscritto

Scrivete un programma che permette di trovare il raggio di una traccia circolare passante attraverso i punti che formano un triangolo. Il raggio (r) del cerchio che inscrive il triangolo è dato dalla formula:

$$r = \frac{a}{2\text{sen}A} = \frac{b}{2\text{sen}B} = \frac{c}{2\text{sen}C}$$

dove a, b, c e A, B, C sono rispettivamente i lati e gli angoli del triangolo.

Il programma è riportato nella tabella A 10, i

risultati, cioè: a = 452m, b = 386m e c = 739 m sono invece indicati nell'Appendice B. (Nota: cos B = $(a^2 + c^2 - b^2)/2ac$.)

Problema 2 - Aree di triangoli

L'area di un triangolo con i lati a, b e c e gli angoli A,B e C, può essere calcolata se sono noti tre lati o due lati e l'angolo compreso usando una delle due formule seguenti:

area del triangolo $=\sqrt{(s(s-a)\ (s-b)\ (s-c))}$

dove 2s = a+b+c

o area = 1/2 absenC o 1/2bcsen A o 1/2acsenB

Scrivete un programma che riporti le aree dei triangoli della Tabella 7.7 usando le intestazioni adatte. Ricordate che i valori sconosciuti devono essere fissati a zero. Fate in modo che tutte le aree vengano calcolate in centimetri quadrati con numeri arrotondati alla prima cifra decimale. Il programma è riportato nella Tabella A11, i risultati nell'appendice B.

Lati del triangolo (cm)			Angolo compreso
a	*b*	*c*	*(in gradi)*
17.2	9.8	14.1	0
0	74	98	125.4
292	0	405	30.5
10.3	15.6	0	69
−1	0	0	0

Tabella 7.7 Dati usati per calcolare le aree dei triangoli

Problema 3 - Volumi dei solidi

Il volume di un solido con l'area della sezione trasversale (A) uniforme e di altezza (H) è dato dalla formula:

V = A x H

Nella tabella 7.8 sono riportate, assieme ai loro codici, le aree delle sezioni trasversali dei solidi più comuni.

Codice	Solido	Area sezione trasversale
1	cubo	base x altezza
2	cilindro	$\pi \times r^2$
3	poliedro esagon.	$1/2\ \sqrt{27} \times r^2$

b = base h = altezza l = lato r = raggio

Tabella 7.8

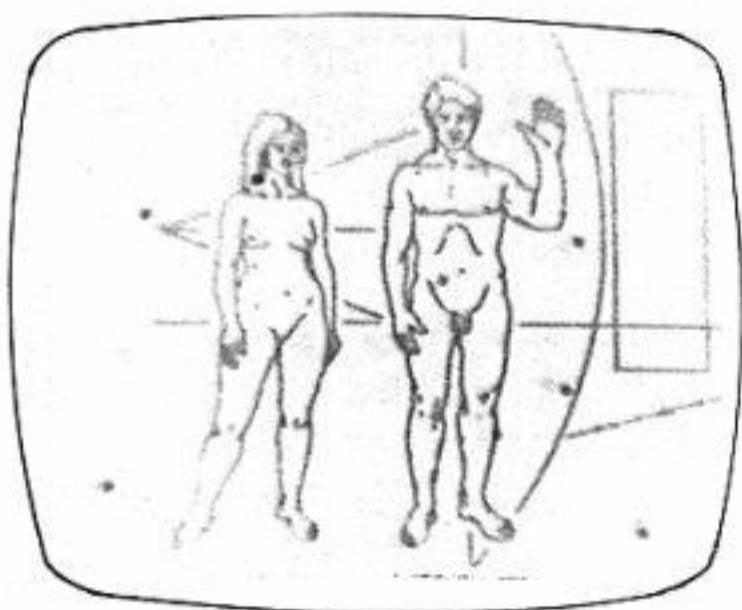
Scrivete un programma che calcoli i volumi di questi solidi, in modo che tutte le dimensioni vengano espresse in centimetri. Il nome del solido va indicato nell'istruzione DATA in modo che il programma generi assieme al volume il nome del solido cui esso si riferisce. Per porre fine all'esecuzione del programma, potete usare il codice 0. Per avere dei codici di prova adatta, attenetevi a quanto riportato nella tabella 7.9, ricordando comunque che potete aggiungere anche dati di vostra scelta.

Codice	Prima dimensione *(b, l, r)*	Seconda dimens. *(h o zero)*	Altez. *(h)*	N. di cifre dec.
2	4.5	0	1.75	2
3	12.6	0	250	0
1	5.3	7.0	4.2	1

Tabella 7.9 Dati per il problema "Volumi di solidi"

Il programma permette all'utente di arrotondare il risultato ad un certo numero di cifre decimali usando un fattore di scala (F) (vedi il paragrafo dal titolo "Arrotondamento").

Vuoi partecipare a Tandem?

"Commodore Computer Club" invita i lettori a partecipare alla popolare trasmissione della Rai Due con videogiochi nuovi ed originali. Ecco come fare.

RAI, Rete 2, ore 14,35. Nessuno spot, niente pubblicità nè TG speciali, ma Tandem. Una trasmissione di giochi, attualità ed ospiti che, per il suo svolgimento, ci richiama più alla memoria i banchi di scuola che Mike Buongiorno. Una trasmissione che va in onda tutti i giorni dal lunedì al venerdì, e che può contare su di un alto indice di telespettatori, non solo giovanissimi, che seguono, partecipano e giocano a distanza secondo la riuscitissima regola del coinvolgere attivamente il pubblico a casa. Una trasmissione collaudata, visto che utilizza, in combinata con altri games, il "Paroliamo" che è già andato in onda con successo, qualche anno fa, su Tele Montecarlo condotto da Lea Pericoli.

Tandem è condotto quotidianamente da Claudio Sorrentino ed è curato da Ettore Desideri. Il pubblico presente in sala, è composto da ragazzi di scuole medie superiori ed i contendenti sono i "capoclasse" che si sfidano a nome della classe o dell'istituto al gioco del Paroliamo. Una sfida alla quale può intervenire, annunciato da un segnale, un computer. I ragazzi scelgono una serie di consonanti e di vocali estratte a sorte e con queste tentano di comporre la parola, aggettivo o sostantivo, più lunga possibile. Parola che, naturalmente, deve avere senso ed essere presente nel vocabolario della lingua italiana, severamente consultato dal giudice Marco Danè. I giovani concorrenti spremono le meningi e sul computer gira il programma di autoistruzione.

E' sempre seguendo il concetto dell'interattività che su RAI 2, nella trasmissione Tandem, sono stati inseriti altri interessanti giochi quali "T.P." e "RAIDUE". Vediamo di che cosa si tratta.

"Telephone Pole" (T.P.), accompagna tutte le trasmissioni di Tandem e grazie ad una centralina telefonica computerizzata è possibile effettuare sondaggi in diretta con il pubblico a casa. All'inizio del gioco-sondaggio, il presentatore annuncia un quesito; uno di questi è stato: "Esiste, secondo voi, la vita nello spazio? ".

I telespettatori, con l'ausilio del telefono, chiamano i numeri indicati e comunicano alla voce sintetizzata del Telephone Pole la scelta personale all'opzione che il computer ha presentato.

Con "d'accordo-contrario" si determina l'andamento del sondaggio che, per ora, può avvenire mediante 15 linee telefoniche entranti in contemporanea. Per ora; dato che "T.P." è una macchina modulare e, quindi, estendibile ad un potenziale utilizzo molto più grande, addirittura un centinaio di linee telefoniche. Presto arriverà anche un chip, che permetterà alla macchina di ampliare la gamma del quesito, estendendo la richiesta a quattro elementi fra i quali sarà possibile scegliere rispondendo, per esempio, con A - B - C o D.

La necessità di ampliare il "Telephone Pole" non è dovuta solamente al desiderio di far giocare più persone nello stesso arco di tempo in cui è possibile parte-

cipare al sondaggio, ma la validità del sondaggio stesso è direttamente proporzionale al numero di chiamate ricevute. Il che equivale al numero di linee telefoniche allacciate, per evitare che questa parte del programma perda la validità peculiare della diretta. In effetti, non esiste altro tipo di questionario collettivo in tempo reale.

I dati raccolti dal "Telephone Pole" vengono subito annunciati dal presentatore che è così in grado di confermare l'esito del sondaggio.

Altro intervento diretto, che è possibile instaurare con Tandem nell'edizione del giovedì, viene dato dall'opportunità di giocare con il videogame "Raidue" ed il telefono. Al via, sul teleschermo, compare l'astronave E.T. che manifesta la chiara intenzione di minacciare la sicurezza della vita sulla Terra. A noi terrestri non resta che la possibilità di difenderci con l'ausilio del computer, che governa i controlli dei cannoni-laser. Telefonando ai numeri indicati sul teleschermo e pronunciando la micidiale parola di codice "Raidue", daremo il via ai colpi sparati dai nostri cannoni-laser, cercando di eliminare il maggior numero possibile di Extra Terrestri. E' la voce dell'utente al telefono che si sostituisce ai tradizionali comandi joystick o paddle, ed il videogame si fa grande, a livello nazionale, grazie alla trasmissione via etere. Unico neo è che, se i fortunati possessori di videogiochi personali possono essere tanti, ma isolati, i contendenti nella battaglia "Raidue" non possono essere altro che due per volta, i primi, quindi, che hanno occupato le linee telefoniche o i primi che hanno raggiunto le postazioni di difesa SIP al messaggio di invasione da parte di Extra Terrestri.

Il venerdì a Tandem arriva "Invideo News", sette minuti di immagini e musica (e che immagini e che musica!) dedicati ogni volta a videogiochi, Commodore in gran parte, applicazioni dell'elettronica nel campo delle telecomunicazioni, software o messaggi spaziali. Ovviamente, questi sono i punti che verranno sviluppati nelle varie puntate di "Invideo News". E siccome, dice un comunicato, "non è facile effettuare e capire gli effetti dell'informatica sulla vita personale e sociale dell'uomo", e pertanto risulterebbe altrettanto difficile "fornirne una descrizione sistematica", si è preferito dare una serie di informazioni sfaccettate, raccolte ed unite casualmente, ponendo particolare attenzione all'esito grafico ed alla somma immagine statica-immagine in movimento, abbinandole al suono, per dare la sensazione della spettacolarità o rendere nel più affascinante dei modi le informazioni, che provengono dal nuovo mondo dell'informatica e dei video-games.

Abili realizzatori di questa parte del programma di Tandem, che va sotto il titolo di "Invideo News", sono i ragazzi dello Studio Pontaccio, che sotto l'influenza di un regista affascinato quanto un ragazzino dalle possibilità offerte dal computer e dal mezzo televisivo, hanno concentrato in questi sette minuti una grande quantità di notizie utili, affascinanti ed interessanti, quanto misteriose.

Ma torniamo un po' indietro nel palinsesto settimanale di Tandem. Il martedì, è l'attività ludica che fa spettacolo. Dei ragazzi sono invitati in studio a gareggiare tra loro con i videogames disponibili. La Commodore ha messo a disposizione di Tandem numerosi programmi per il Vic 20, ed i ragazzi si dimostrano espertissimi conoscitori di battaglie ed atterraggi spaziali, da effettuare con questi sistemi.

Anche Commodore Computer Club vuole partecipare a Tandem, ma, essendo una rivista, non può che invitare i suoi lettori a partecipare alla trasmissione televisiva della RAI 2! Allora forza, al lavoro. Inviando in redazione i programmi dei giochi che avete realizzato per una qualsiasi macchina Commodore, potrete essere invitati a dar prova della vostra abilità negli studi della TV.

Una commissione costituita dalla nostra redazione, per l'aspetto tecnico dei programmi, e dallo Studio Pontaccio, per valutare l'effetto grafico e la "telegenia" del videogame, vaglierà la qualità dei prodotti da voi inviati. La fama sarà il giusto premio.

Allora, o vi sintonizzate ogni giorno su RAI 2, ore 14,35, TANDEM o siete fuori. ▪

Pinuccio Biselli

Un programma completo per il word processing

Non era mai successo. Nessuna rivista è mai riuscita a regolare ai suoi lettori un programma professionale per l'elaborazione dei testi. Commodore Computer Club sì!

```
10 POKE59468,14:PRINT"";:PRINT" |";:FORN=2TO39:PRINT"-";:NEXTN:PRINT"| ";
20 FORN=2TO23:PRINT" |";SPC(38);"| ";:NEXTN
30 PRINT" |";:FORN=2TO39:PRINT"_";:NEXTN:PRINT"| ";
40 PRINT"";TAB(5)"o O R D   ¬R O C E S S I N G"
50 PRINT:PRINTTAB(15)"* * * * *"
60 PRINT:PRINT
70 PRINT:PRINTTAB(6)"COPYRIGHT 1981 BITS & BYTES"
80 PRINTTAB(5)"VIALE MONTE GRAPPA 4 - MILANO"
90 PRINTTAB(12)"TEL. 02 - 573344"
95 FORT=1TO3000:NEXTT
100 CLR:DIMA$(150)
101 FORX1=1TO40:X1$=X1$+"-":NEXTX1
102 X2$=""
103 FORX1=1TO40:X3$=X3$+" ":NEXTX1:FORX1=1TO40:X3$=X3$+"":NEXTX1
104 X4$=CHR$(13)
110 PRINT"ELEZIONE FUNZIONI:":PRINTX1$;
120 PRINT"C=CREAZIONE TESTO":PRINT:PRINT"P=PRELIEVO TESTO DA NASTRO"
130 GETZ$:IFZ$="C"THEN1000
131 IFZ$="P"THEN2000
132 GOTO130
1000 PRINT"REAZIONE TESTO"
1010 PRINTX1$;LEFT$(X2$,15);X1$;LEFT$(X2$,22);X1$;
1020 N=1
1100 IFN>140ORFRE(0)<1000THENGOSUB5100:GOTO1200
1101 GOSUB4000
1110 IFLEN(B$)<37THEN1140
1120 GOSUB4200
1130 A$(N)=G$:N=N+1:GOTO1110
1140 A$(N)=B$:N=N+1:IFRIGHT$(A$(N-1),1)="@"THEN1600
1150 GOSUB4300
1160 GOSUB5000
1200 PRINTLEFT$(X2$,23);"P=PROSEGUIMENTO"
1201 PRINT"L=LISTING";SPC(22);"E=EDITING";
1210 GETS1$:IFS1$="P"ORS1$="L"ORS1$="E"THEN1212
1211 GOTO1210
1212 PRINTLEFT$(X2$,23);X3$;"";X3$;
1220 IFS1$="P"THEN1100
1221 IFS1$="L"THEN1230
1222 GOTO1300
1230 IFN<2THEN1200
1240 GOSUB4900:GOTO1200
```

```
1300 IFN<=1THEN1200
1301 PRINTLEFT$(X2$,23);"▓I■=INSERIMENTO";SPC(15);"▓C■=CORREZIONE";
1302 PRINT"▓E■=ELIMINAZIONE";SPC(12);"▓F■=FINE EDITING";
1310 GETS3$:IFS3$="I"ORS3$="C"ORS3$="E"ORS3$="F"THEN1312
1311 GOTO1310
1312 PRINTLEFT$(X2$,23);X3$;"▓";X3$;
1320 IFS3$="I"THEN1330
1321 IFS3$="C"THEN1440
1322 IFS3$="E"THEN1480
1323 GOTO1200
1330 IFN>140ORFRE(0)<1000THENGOSUB5100:GOTO1300
1331 IFN>2THENPRINTLEFT$(X2$,23);X3$;"-OPO QUALE RIGA ? ";:GOTO1340
1332 GOTO1300
1340 GOSUB4100:R1=VAL(B$):IFR1<=0ORR1>=N-1THEN1330
1341 PRINTLEFT$(X2$,23);X3$;:GOSUB4300
1350 YY=0:FORM=R1TO1STEP-1:YY=YY+1
1351 PRINTLEFT$(X2$,15-YY);SPC(4-LEN(STR$(M)));MID$(STR$(M),2);" ";
1352 PRINTA$(M);:IFYY=12THEN1360
1353 NEXTM
1360 GOSUB4000:M=1
1370 IFLEN(B$)<37THEN1400
1380 GOSUB4200
1390 H$(M)=G$:M=M+1:GOTO1370
1400 H$(M)=B$:M=M+1
1410 IFRIGHT$(H$(M-1),1)="@"THENGOSUB4500:GOTO16
1411 GOSUB4400
1420 GOSUB4300:IFN>1THENGOSUB4800:GOTO1430
1421 GOTO1100
1430 R0=0:GOTO1300
1440 IFN>140ORFRE(0)<1000THENGOSUB5100:GOTO1300
1441 IFN>1THENGOSUB4600:GOTO1450
1442 GOTO1300
1450 YY=0:FORM=R1TO1STEP-1:YY=YY+1
1451 PRINTLEFT$(X2$,15-YY);SPC(4-LEN(STR$(M)));MID$(STR$(M),2);" ";
1452 PRINTA$(M);:IFYY=12THEN1460
1453 NEXTM
1460 GOSUB4700
1470 R1=R1-1:R0=1:GOTO1360
1480 IFN>1THENGOSUB4600:GOTO1490
1481 GOTO1300
1490 GOSUB4700:R1=R1-1:R0=0:IFR1=0THENR1=1
1500 IFN>1THENGOSUB4800
1510 GOTO1300
1600 GOSUB4300
1610 GOSUB5000
1620 IFRR=1THEN1630
1621 PRINTLEFT$(X2$,23);"▓L■=LISTING"
1622 PRINT"▓F■=FINE CREAZIONE TESTO";SPC(9);"▓R■=RIPRESA";
1623 GOTO1640
1630 PRINTLEFT$(X2$,23);"▓L■=LISTING"
1631 PRINT"▓F■=FINE REVISIONE TESTO";SPC(9);"▓R■=RIPRESA";
1640 GETS4$
1641 IFS4$="L"ORS4$="F"ORS4$="R"THENPRINTLEFT$(X2$,23);X3$;"▓";X3$;:GOTO1650
1642 GOTO1640
1650 IFS4$="L"THENGOSUB4900:GOTO1620
1651 IFS4$="F"THEN1700
1660 A$(N-1)=LEFT$(A$(N-1),LEN(A$(N-1))-1):IFA$(N-1)=""THENN=N-1
1670 GOSUB4300:IFN>1THENGOSUB5000
1680 GOTO1200
1700 PRINT"▶ELEZIONE SUB-FUNZIONI:":PRINTX1$;
```

```
1710 PRINT"█S■=STAMPA TESTO":PRINT:PRINT"█R■=REGISTRAZIONE TESTO SU NASTRO"
1711 PRINT:PRINT"█V■=REVISIONE TESTO":PRINT
1712 PRINT"█N■=NUOVA ESECUZIONE PROGRAMMA":PRINT:PRINT"█F■=FINE PROGRAMMA"
1720 GETZ$:IFZ$="S"THEN2100
1721 IFZ$="R"THEN1800
1722 IFZ$="V"THEN1900
1723 IFZ$="N"THEN100
1724 IFZ$="F"THEN3000
1725 GOTO1720
1800 PRINT"□\NTRODURRE I SEGUENTI DATI:":PRINTX1$;
1801 PRINT"/OME DEL PRESENTE TESTO ";
1802 GOSUB4100:FF$=B$:IFFF$=""ORLEN(FF$)>10THENGOSUB5920:GOTO1802
1810 OPEN1,1,1,FF$
1820 PRINT"□█_EGISTRAZIONE IN CORSO■";
1830 FORM=1TON-1
1840 B$=CHR$(161)+A$(M)
1850 FORI=1TOLEN(B$)
1851 IFMID$(B$,I,1)=","THENB$=LEFT$(B$,I-1)+CHR$(162)+MID$(B$,I+1)
1860 NEXTI
1865 FORI=1TOLEN(B$)
1866 IFMID$(B$,I,1)=":"THENB$=LEFT$(B$,I-1)+CHR$(163)+MID$(B$,I+1)
1867 NEXTI
1870 PRINT#1,B$;X4$;:NEXTM
1880 CLOSE1:GOTO1700
1900 PRINT"□_EVISIONE TESTO"
1910 PRINTX1$;LEFT$(X2$,15);X1$;LEFT$(X2$,22);X1$;
1920 RR=1:GOTO1630
2000 PRINT"□\NTRODURRE I SEGUENTI DATI:":PRINTX1$;
2001 PRINT"/OME DEL TESTO DESIDERATO ";
2002 GOSUB4100:FF$=B$:IFFF$=""ORLEN(FF$)>10THENGOSUB5920:GOTO2002
2010 OPEN1,1,0,FF$
2020 PRINT"□█RELIEVO DA NASTRO IN CORSO■";
2030 N=N+1
2040 INPUT#1,B$:B$=MID$(B$,2)
2050 FORI=1TOLEN(B$)
2051 IFMID$(B$,I,1)=CHR$(162)THENB$=LEFT$(B$,I-1)+","+MID$(B$,I+1)
2060 NEXTI
2065 FORI=1TOLEN(B$)
2066 IFMID$(B$,I,1)=CHR$(163)THENB$=LEFT$(B$,I-1)+":"+MID$(B$,I+1)
2067 NEXTI:A$(N)=B$
2070 IFRIGHT$(A$(N),1)="@"THENN=N+1:CLOSE1:GOTO1700
2071 GOTO2030
2100 PRINT"□-EFINIRE I SEGUENTI PARAMETRI DI STAMPA:";:PRINTX1$;
2110 PRINTLEFT$(X2$,3);X3$;"_ORMATO VERT. FOGLIO (IN RIGHE)    ";
2111 GOSUB4100:V1=VAL(B$):IFV1<1THENGOSUB5920:GOTO2111
2120 PRINTLEFT$(X2$,5);X3$;"/UMERO MAX. RIGHE PER FOGLIO       ";
2121 GOSUB4100:V2=VAL(B$):IFV2<1ORV2>V1THENGOSUB5920:GOTO2121
2130 PRINTLEFT$(X2$,7);X3$;"\NTERLINEA (1-2-3)                 ";
2131 GOSUB4100:V3=VAL(B$)
2132 IFV3<>1ANDV3<>2ANDV3<>3ORV2*V3>V1THENGOSUB5920:GOTO2131
2140 PRINTLEFT$(X2$,9);X3$;"/UMERO RIGA INIZIO STAMPA          ";
2141 GOSUB4100:V4=VAL(B$):IFV4<1ORV4>V1-V2*V3+1THENGOSUB5920:GOTO2141
2142 V4=V4-1
2160 PRINTLEFT$(X2$,11);X3$;"_ORMATO ORIZZ. FOGLIO (IN CAR.)   ";
2161 GOSUB4100:O1=VAL(B$):IFO1<1ORO1>132THENGOSUB5920:GOTO2161
2162 O1=O1-1
2170 PRINTLEFT$(X2$,13);X3$;"/UMERO MAX. CARATTERI PER RIGA     ";
2171 GOSUB4100:O2=VAL(B$):IFO2<1ORO2>O1THENGOSUB5920:GOTO2171
2180 PRINTLEFT$(X2$,15);X3$;"\ARGINATORE SINISTRO (IN SPAZI)    ";
2181 GOSUB4100:O3=VAL(B$):IFB$=""ORO3<0ORO3>O1-O2THENGOSUB5920:GOTO2181
```

```
2190 PRINTLEFT$(X2$,16);X1$;:PRINT"-ORREZIONI (S-N) ";
2191 GOSUB4100:IFB$="S"THENPRINTLEFT$(X2$,17);X3$;:GOTO2110
2192 IFB$<>"N"THENGOSUB5920:GOTO2191
2200 PRINT"]♥CEGLIERE IL TIPO DI STAMPA:":PRINTX1$;
2210 PRINTLEFT$(X2$,3);"N=NORMALE  V=VIVO  S=SIMMETRICA ";
2211 GOSUB4100:IFB$="N"THENW1=1:GOTO2215
2212 IFB$="V"THENW1=2:GOTO2215
2213 IFB$="S"THENW1=3:GOTO2215
2214 GOSUB5920:GOTO2211
2215 PRINTLEFT$(X2$,5);"-ODIFICA MINUSCOLE (S-N) ";
2216 GOSUB4100:IFB$="S"THENMK=1:GOTO2220
2217 IFB$="N"THENMK=0:GOTO2220
2218 GOSUB5920:GOTO2216
2220 PRINT"]♥PECIFICARE NUMERO COPIE DA STAMPARE:":PRINTX1$;
2230 PRINTLEFT$(X2$,3);"/UMERO COPIE ";
2231 GOSUB4100:W3=VAL(B$):IFW3<1THENGOSUB5920:GOTO2231
2240 PRINT"]♥PECIFICARE SE IL TESTO PREVEDE"
2241 PRINT"IL PRELIEVO DELL'INDIRIZZO DA NASTRO":PRINT"O DA TASTIERA:"
2242 PRINTX1$;
2250 PRINTLEFT$(X2$,5);"S=SI  N=NO ";
2251 GOSUB4100:IFB$="N"THENW4=0:GOTO2260
2252 IFB$<>"S"THENGOSUB5920:GOTO2251
2255 PRINTLEFT$(X2$,5);X3$;"N=DA NASTRO  T=DA TASTIERA ";
2256 GOSUB4100:IFB$="N"THENW4=1:GOTO2260
2257 IFB$="T"THENW4=2:GOTO2260
2258 GOSUB5920:GOTO2256
2260 PRINT"]PTARE PER L'AVANZAMENTO A FINE FOGLIO:":PRINTX1$;
2270 PRINTLEFT$(X2$,3);"M=MANUALE  A=AUTOMATICO ";
2271 GOSUB4100:IFB$="M"THENW2=1:GOTO2280
2272 IFB$="A"THENW2=2:GOTO2280
2273 GOSUB5920:GOTO2271
2280 PRINT"]\NTRODURRE I SEGUENTI DATI:":PRINTX1$;
2281 IFW4<>1THEN2292
2282 PRINT"/OME DELL'ELENCO VOLUTO ";
2283 GOSUB4100:FF$=B$:IFFF$=""ORLEN(FF$)>10THENGOSUB5920:GOTO2283
2290 OPEN1,1,0,FF$:PRINT:PRINT
2292 PRINT"|IPO DI STAMPANTE (CBM-HON) ";
2293 GOSUB4100:P$=B$
2294 IFP$="CBM"THENP=4:GOTO2297
2295 IFP$="HON"THENP=5:GOTO2297
2296 GOSUB5920:GOTO2293
2297 OPEN2,P:IFP=4THENOPEN7,4,7
2298 IFMK=1THENGOSUB5950
2300 F$="":PRINT"]♥TAMPA IN CORSO■";
2310 FORQ2=1TOW3:N=0:D$="":W5=0:PRINTLEFT$(X2$,4);"-OPIA NO.";Q2;
2311 IFW4<>0THENONW4GOSUB5700,5800
2320 GOSUB5300
2330 N=N+1:A$=A$(N):IFRIGHT$(A$,1)<>"#"THEN2340
2331 W5=W5+1:A$=LEFT$(A$,LEN(A$)-1)+J$(W5)+CHR$(185):J$(W5)="":IFW5=4THENW5=0
2340 IFLEN(D$)>0THENA$=D$+A$:D$=""
2350 L1=LEN(A$):IFL1<=O2+1THEN2400
2351 GOTO2500
2400 IFRIGHT$(A$,1)=CHR$(185)THENF$=A$:GOSUB5500:A$="":GOSUB5200:GOTO2430
2410 IFRIGHT$(A$,1)<>"@"THEN2420
2411 F$=A$:GOSUB5500:A$="":GOSUB5200:GOSUB5400:NEXTQ2:N=N+1
2412 CLOSE1:CLOSE2:IFP=4THENCLOSE7
2413 GOSUB6000:GOTO1700
2420 IFL1<>O2ORRIGHT$(A$,1)=CHR$(185)ORRIGHT$(A$,1)="@"THEN2430
2421 F$=A$:GOSUB5500:A$="":GOSUB5200
2430 IFQ1=V2*V3+V4THENGOSUB5400:GOSUB5300
```

Scrivi anche tu

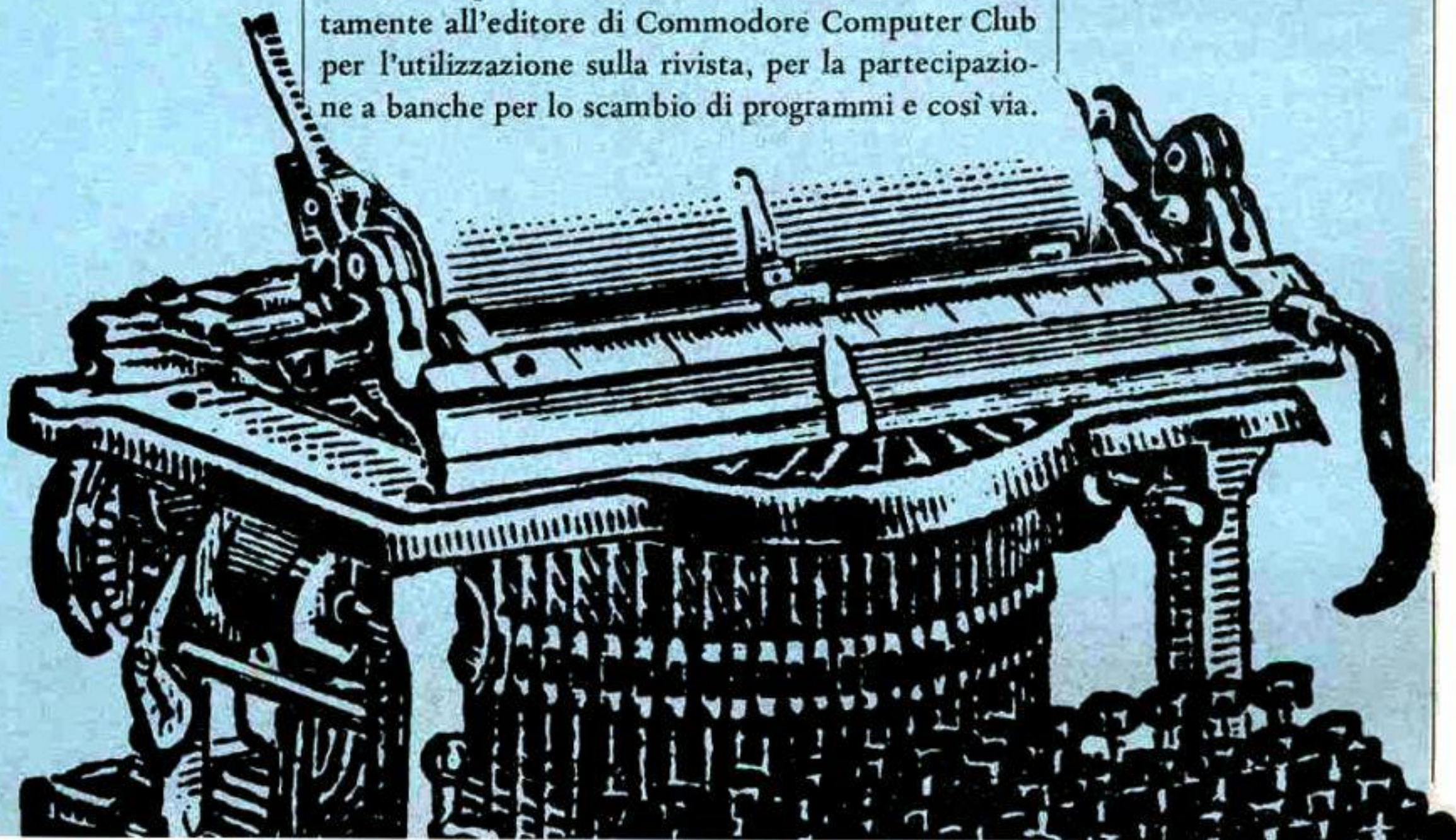
La collaborazione a Commodore Computer Club è aperta a tutti gli utenti di sistemi CBM che abbiano sviluppato programmi e routine originali e d'interesse generale o che desiderino fare partecipi gli altri lettori delle scoperte e delle osservazioni fatte per una migliore utilizzazione di queste macchine.

Allegate un breve articolo in cui spiegate a che serve il programma, come funziona, su quale computer gira e con quale configurazione (espansa, non espansa, ecc.).

- Iniziate tutti i listati con delle Rem indicanti il titolo del programma, il nome, l'indirizzo ed il telefono dell'autore.
- Inviate i listati su disco o su cassetta, (su carta verranno irrimediabilmente cestinati).

Per compensarvi per questa spesa la redazione vi spedirà a stretto giro di posta due cassette e/o dischetti nuovi per ciascuna cassetta da voi inviata, indipendentemente dal fatto che il programma registrato venga accettato per la pubblicazione.

Ciascun articolo e/o programma pubblicato, invece, verrà compensato con l'invio d'una intera scatola di cassette e/o dischetti.

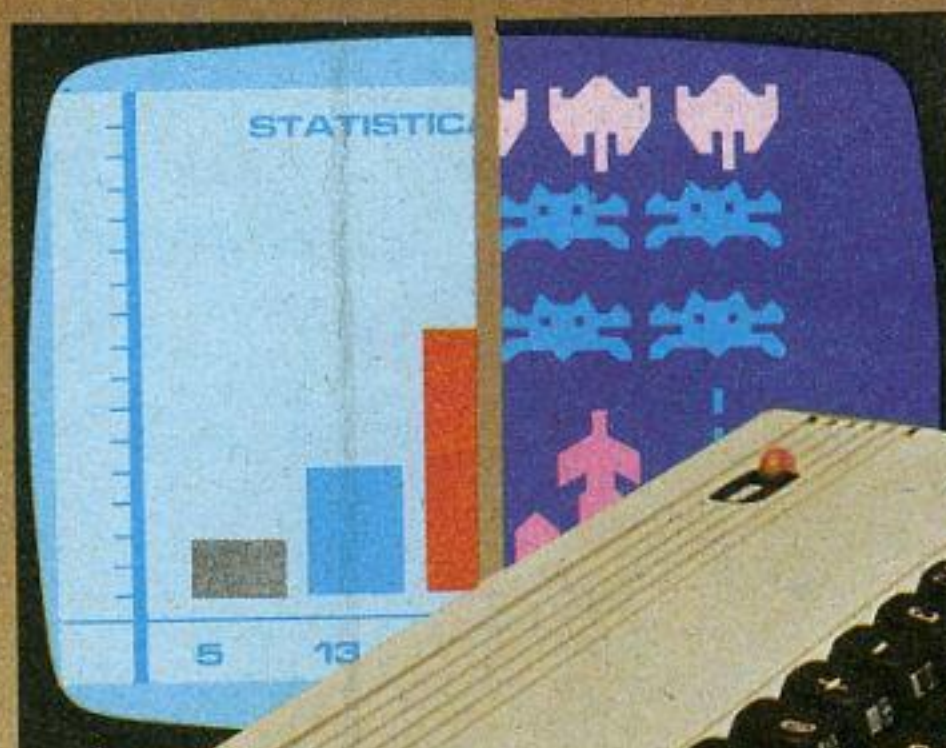
Gli autori, naturalmente, si assumono ogni responsabilità circa l'originalità dei lavori inviati i cui diritti di riproduzione si intendono ceduti illimitatamente all'editore di Commodore Computer Club per l'utilizzazione sulla rivista, per la partecipazione a banche per lo scambio di programmi e così via.

SE VUOI ESSE
DI SCEGLIERE

Ogni giorno in Europa si apre un nuovo computer shop.
Un pubblico sempre più numeroso è attirato verso il personal e si rivolge ai negozi specializzati per trovare la sua marca preferita.
Tu che hai capito qual è il futuro dei computer e hai deciso di aprire un negozio, cerchi un nome che dia prestigio e una organizzazione che non ponga vincoli ma offra vantaggi concreti.
Computeria vuol dire negozi di computer fin dal 1979.
Computeria è anche una organizzazione che ha avviato rapporti di collaborazione con tutti i principali fornitori, perciò i suoi affiliati possono scegliere e vendere le marche più prestigiose e richieste.
E inoltre Computeria ti dà un prezioso know-how, una ricchissima dotazione di programmi, supersconti esclusivi, vantaggi economici sul leasing.
E tanta pubblicità.
Se vuoi essere libero di scegliere quello che vuoi vendere nel tuo negozio, l'organizzazione Computeria è la tua scelta obbligata.

COMPUTERIA®

La catena senza catene.

RE LIBERO

VIC 20
"COMPUTER DELL'ANNO"

VIDEOGIOCHI
disponibili varie cassette videogame.

PERIFERICHE
si possono collegare direttamente registratore, dedicato, floppy, stampante, joystick.

TASTIERA
professionale con 66 tasti, 256 combinazioni di colori, 512 segni diversi per creare grafici, disegni, videogiochi.

SOFTWARE
una vasta gamma di programmi per tutte le necessità.

SUONO
3 toni e 9 ottave disponibili per creare le tue musiche ed effetti speciali.

VIC 20 è il computer più diffuso nel mondo, oltre un milione di esemplari venduti. Perchè?
1. VIC 20 è facile da usare: lo colleghi alla TV e premi i tasti. Tutto qui. - 2. Ci puoi giocare: dagli scacchi agli invasori spaziali, e puoi inventarti i tuoi giochi.
3. Lo usi per la scuola, per risolvere problemi e compiti. - 4. Lo impieghi in casa, per la contabilità entrate-uscite, ma anche per le ricette e per il bioritmo...
5. Costa solo 423.000 lire più iva. - 6. Puoi imparare a programmare in Basic, la lingua del futuro.
7. È stato scelto come "computer dell'anno" dalle migliori riviste internazionali di informatica.
8. È della Commodore, la N° 1 in Europa e in Italia nel campo dei Personal Computer.
Perchè non farti un regalo intelligente?

ETHOS PISA

```
2440 IFLEN(A$)>0THEND$=A$:A$="":IFRIGHT$(D$,1)<>" "THEND$=D$+" "
2450 GOTO2330
2500 IFO2>=L1THEN2350
2510 FORI=O2+1TO2STEP-1:C$=MID$(A$,I,1):IFC$=" "ANDMID$(A$,I-1,1)<>" "THEN2530
2511 IFC$="!"ORC$=")"ORC$=","ORC$="."ORC$=":"ORC$=";"ORC$="?"THEN2513
2512 GOTO2514
2513 IFI<O2+1THEN2520
2514 NEXTI:I=O2
2520 H$=LEFT$(A$,I):D$=MID$(A$,I+1):A$="":GOTO2540
2530 H$=LEFT$(A$,I-1):D$=MID$(A$,I+1):A$=""
2540 IFLEFT$(D$,1)=" "THEND$=MID$(D$,2):GOTO2540
2550 F$=H$:GOSUB5500:A$="":GOSUB5200
2560 IFQ1=V2*V3+V4THENGOSUB5400:GOSUB5300
2570 GOTO2340
3000 POKE59468,12:PRINT"JFINE PROGRAMMA":END
4000 P1=0:PRINTLEFT$(X2$,16);:GOSUB4100:IFB$=""THENB$=CHR$(172)
4010 IFLEFT$(B$,1)<>CHR$(62)THEN4030
4011 P1=VAL(MID$(B$,2))
4012 IFP1<1ORP1>120THEN4020
4013 E$="":FORX1=1TOP1:E$=E$+" ":NEXTX1
4014 PRINTLEFT$(X2$,16);X3$;E$;:GOSUB4100:B$=E$+B$:GOTO4030
4020 PRINTLEFT$(X2$,16);X3$;LEFT$(X2$,23);"* MARGINATORE ERRATO■ *";
4021 FORT=1TO2000:NEXTT:PRINTLEFT$(X2$,23);X3$;:GOTO4000
4030 RETURN
4100 B$="":Y=0
4110 PRINT"_";
4120 GETC$:IFC$<>""THENIFASC(C$)=34THEN4120
4121 IFC$=CHR$(160)THENC$=" "
4122 IFC$=CHR$(141)THENC$=CHR$(13)
4123 IFLEN(B$)=239-P1ANDC$<>CHR$(20)ANDC$<>CHR$(13)ORC$=""THEN4120
4130 IFC$=CHR$(20)ANDY=0THENPRINT"▌";:GOTO4100
4131 IFC$=CHR$(20)THEN4180
4140 IFC$=CHR$(13)THENPRINT"▌ ";:GOTO4190
4150 IFC$=CHR$(29)ORC$=CHR$(62)THENPRINT"▌>";:B$=B$+CHR$(62):Y=Y+1:GOTO4110
4160 IFC$=CHR$(17)THENPRINT"▌L";:B$=B$+CHR$(185):Y=Y+1:GOTO4190
4170 PRINT"▌";C$;:B$=B$+C$:Y=Y+1:IFC$<>"@"ANDC$<>"#"THEN4110
4171 GOTO4190
4180 PRINT"▌▌  ▌▌";:B$=LEFT$(B$,LEN(B$)-1):Y=Y-1:GOTO4110
4190 RETURN
4200 FORI=36TO1STEP-1:F$=MID$(B$,I,1)
4201 IFF$=" "ORF$="!"ORF$=")"ORF$=","ORF$="."ORF$=":"ORF$=";"ORF$="?"THEN4210
4202 NEXTI:I=36
4210 G$=LEFT$(B$,I):B$=MID$(B$,I+1)
4220 RETURN
4300 FORXX=3TO14:PRINTLEFT$(X2$,XX);X3$;:NEXTXX
4301 FORXX=16TO21:PRINTLEFT$(X2$,XX);X3$;:NEXTXX
4310 RETURN
4400 FORMM=N-1TOR1+1STEP-1:A$(MM+M-1)=A$(MM):NEXTMM
4410 FORMM=1TOM-1:A$(R1+MM)=H$(MM):NEXTMM:N=N+M-1
4420 RETURN
4500 FORMM=1TOM-1:A$(R1+MM)=H$(MM):NEXTMM
4510 IFN-1>=R1+MTHENFORMM=R1+MTON-1:A$(MM)="":NEXTMM
4520 N=R1+M
4530 RETURN
4600 PRINTLEFT$(X2$,23);X3$;"●UALE RIGA ? ";
4610 GOSUB4100:R1=VAL(B$):IFR1<=0ORR1>N-1THEN4600
4611 PRINTLEFT$(X2$,23);X3$;:GOSUB4300
4620 RETURN
4700 FORMM=R1TON-1:A$(MM)=A$(MM+1):NEXTMM:N=N-1
```

```
4710 RETURN
4800 YY=0:FORM=R1+R0TON-1:YY=YY+1
4801 PRINTLEFT$(X2$,YY+2);SPC(4-LEN(STR$(M)));MID$(STR$(M),2);" ";
4802 PRINTA$(M);:IFYY=12THEN4810
4803 NEXTM
4810 RETURN
4900 GOSUB4300:YY=0:FORM=1TON-1
4910 YY=YY+1:PRINTLEFT$(X2$,YY+2);SPC(4-LEN(STR$(M)));MID$(STR$(M),2);" ";
4911 PRINTA$(M);:IFYY=12THEN4920
4912 NEXTM:GOTO4960
4920 IFM=N-1THEN4960
4930 PRINTLEFT$(X2$,23);"█L█=PROSEGUIMENTO LISTING":PRINT"█F█=FINE LISTING";
4940 GETS2$:IFS2$="L"ORS2$="F"THENPRINTLEFT$(X2$,23);X3$;"█";X3$;:GOTO4950
4941 GOTO4940
4950 IFS2$="L"THENGOSUB4300:YY=0:NEXTM
4960 RETURN
5000 YY=0:FORM=N-1TO1STEP-1:YY=YY+1
5001 PRINTLEFT$(X2$,15-YY);SPC(4-LEN(STR$(M)));MID$(STR$(M),2);" ";
5002 PRINTA$(M);:IFYY=12THEN5010
5003 NEXTM
5010 RETURN
5100 PRINTLEFT$(X2$,23);"* █\EMORIA DISPONIBILE ESAURITA█ *";
5101 FORT=1TO2000:NEXTT:PRINTLEFT$(X2$,23);X3$;
5110 PRINTLEFT$(X2$,23);"█F█=FINE CREAZIONE TESTO":PRINT"█R█=RIPRESA";
5120 GETS5$:IFS5$="F"ORS5$="R"THENPRINTLEFT$(X2$,23);X3$;"█";X3$;:GOTO5130
5121 GOTO5120
5130 IFS5$="F"THENA$(N-1)=A$(N-1)+"@":GOTO1700
5140 RETURN
5200 IFL0=1THEN5220
5201 Q1=Q1+1
5210 IFV3>1THENFORQ=1TOV3-1:PRINT#2," ":Q1=Q1+1:NEXTQ
5220 RETURN
5300 Q1=0:IFV4>0THENFORQ=1TOV4:PRINT#2," ":Q1=Q1+1:NEXTQ
5310 RETURN
5400 IFQ1<V1THENFORQ=Q1+1TOV1:PRINT#2," ":NEXTQ
5410 IFW2=2THEN5450
5420 PRINTLEFT$(X2$,10);"-AMBIARE FOGLIO E PREMERE █▁▔| ⌐╱█ ";
5430 GETZ$:IFZ$=""THEN5430
5431 IFASC(Z$)<>13THEN5430
5440 PRINTLEFT$(X2$,10);X3$;
5450 RETURN
5500 FX$="":IFO2>0THENFORX1=1TOO2:FX$=FX$+" ":NEXTX1
5501 IFF$=FX$THENL0=1:GOTO5680
5510 IFW1<>2THEN5620
5520 L2=0:IFRIGHT$(F$,1)=CHR$(185)ORRIGHT$(F$,1)="@"THEN5522
5521 GOTO5530
5522 F$=LEFT$(F$,LEN(F$)-1):GOTO5610
5530 IFLEN(F$)=O2THEN5610
5540 IFRIGHT$(F$,1)=" "THENF$=LEFT$(F$,LEN(F$)-1):GOTO5540
5550 FORI=1TOLEN(F$):C$=MID$(F$,I,1):IFC$=" "THENL2=L2+1:NEXTI
5560 F$=MID$(F$,L2+1)
5570 L3=0:FORI=1TOLEN(F$)-1:C$=MID$(F$,I,1)
5571 IFC$=" "ORC$="!"ORC$=","ORC$="."ORC$=":"ORC$=";"ORC$="?"THENL3=L3+1
5580 NEXTI:IFL3=0ORLEN(F$)<2THEN5610
5590 FORI=LEN(F$)-1TO1STEP-1:C$=MID$(F$,I,1)
5591 IFC$=" "ANDMID$(F$,I+1,1)<>" "THEN5594
5592 IFC$="!"ORC$=","ORC$="."ORC$=":"ORC$=";"ORC$="?"THEN5594
5593 GOTO5600
5594 F$=LEFT$(F$,I)+" "+MID$(F$,I+1):IFLEN(F$)=O2-L2THEN5610
5600 NEXTI:GOTO5590
```

```
5610 FX$="":IFO3+L2>0THENFORX1=1TOO3+L2:FX$=FX$+" ":NEXTX1
5611 F$=FX$+F$:GOTO5670
5620 IFRIGHT$(F$,1)=CHR$(185)ORRIGHT$(F$,1)="@"THENF$=LEFT$(F$,LEN(F$)-1)
5630 IFW1<>1THEN5640
5631 FX$="":IFO3>0THENFORX1=1TOO3:FX$=FX$+" ":NEXTX1
5632 F$=FX$+F$:GOTO5670
5640 IFLEFT$(F$,1)=" "THENF$=MID$(F$,2):GOTO5640
5650 IFRIGHT$(F$,1)=" "THENF$=LEFT$(F$,LEN(F$)-1):GOTO5650
5660 FX$="":IFO3+INT((O2-LEN(F$))/2)<=0THEN5670
5661 FORX1=1TOO3+INT((O2-LEN(F$))/2):FX$=FX$+" ":NEXTX1:F$=FX$+F$
5670 L0=0:IFF$=""THENF$=" "
5671 PRINT#2,F$
5680 RETURN
5700 INPUT#1,J$(0),J$(1),J$(2),J$(3).J$(4),J$(5)
5710 IFJ$(0)<>"999999"THEN5750
5720 PRINTLEFT$(X2$,10);"* ▩┐LENCO INDIRIZZI ESAURITO■ *";
5721 PRINTLEFT$(X2$,12);"┐ER CONTINUARE,PREMERE ▩─┐ ⁄─⁄■ ";:MM=0
5730 GETZ$:IFZ$=""THEN5730
5731 IFASC(Z$)<>13THEN5730
5740 MM=MM+1:IFRIGHT$(A$(MM),1)<>"@"THEN5740
5741 MM=MM+1:CLOSE1:CLOSE2:IFP=4THENCLOSE7
5742 GOSUB6000:GOTO1700
5750 IFLEN(J$(0))+LEN(J$(1))>=30THEN5752
5751 FORX1=1TO30-LEN(J$(0))-LEN(J$(1)):J$(1)=J$(1)+" ":NEXTX1
5752 J$(1)=J$(1)+J$(0)
5760 J$(4)=J$(4)+" "+J$(5)
5770 RETURN
5800 PRINTLEFT$(X2$,7);"⌐NTRODURRE I SEGUENTI DATI:":PRINTLEFT$(X2$,8);X1$;
5810 PRINTLEFT$(X2$,9);X3$;"▩";X3$;"⊐";"─ODICE ♠//   ";:GOSUB4100:J$(0)=B$
5820 PRINTLEFT$(X2$,11);X3$;"▩";X3$;"⊐";"_IFERIMENTI  ";:GOSUB4100:J$(1)=B$
5830 PRINTLEFT$(X2$,13);X3$;"▩";X3$;"⊐";"⁄OMINATIVO   ";:GOSUB4100:J$(2)=B$
5840 PRINTLEFT$(X2$,15);X3$;"▩";X3$;"⊐";"⌐NDIRIZZO    ";:GOSUB4100:J$(3)=B$
5850 PRINTLEFT$(X2$,17);X3$;"▩";X3$;"⊐";"─ ♠ ┐        ";:GOSUB4100:J$(4)=B$
5860 PRINTLEFT$(X2$,19);X3$;"▩";X3$;"⊐";"LOCALITA'    ";:GOSUB4100:J$(5)=B$
5870 PRINTLEFT$(X2$,21);X1$;:PRINT"─ORREZIONI (S-N) ";
5871 GOSUB4100:IFB$="S"THENPRINTLEFT$(X2$,22);X3$;:GOTO5810
5872 IFB$<>"N"THENGOSUB5920:GOTO5871
5880 IFLEN(J$(0))+LEN(J$(1))>=30THEN5882
5881 FORX1=1TO30-LEN(J$(0))-LEN(J$(1)):J$(1)=J$(1)+" ':NEXTX1
5882 J$(1)=J$(1)+J$(0)
5890 J$(4)=J$(4)+" "+J$(5)
5900 FORX1=7TO22:PRINTLEFT$(X2$,X1);X3$;:NEXTX1
5910 RETURN
5920 IFB$=""THENPRINT"▌ ▌";:GOTO5940
5930 FORX1=1TOLEN(B$):PRINT"▌  ▌▌";:NEXTX1:PRINT"▌ ▌";
5940 RETURN
5950 PRINT"⊐▩─ODIFICA MINUSCOLE IN CORSO■";
5960 FORI=1TON-1
5961 PRINTLEFT$(X2$,4);"_IGA IN ELABORAZIONE:";I;
5970 FY$="":FORX1=1TOLEN(A$(I)):FZ$=MID$(A$(I),X1,1):FX=ASC(FZ$)
5980 IFFX>64ANDFX<91THENFX=FX+32
5990 FY$=FY$+CHR$(FX):NEXTX1:A$(I)=FY$:NEXTI:RETURN
6000 IFMK=0THEN6060
6010 PRINT"⊐▩─ECODIFICA MINUSCOLE IN CORSO■";
6020 FORI=1TON-1
6021 PRINTLEFT$(X2$,4);"_IGA IN ELABORAZIONE:";I;
6030 FY$="":FORX1=1TOLEN(A$(I)):FZ$=MID$(A$(I),X1,1):FX=ASC(FZ$)
6040 IFFX>96ANDFX<123THENFX=FX-32
6050 FY$=FY$+CHR$(FX):NEXTX1:A$(I)=FY$:NEXTI
6060 RETURN
```

Assembler per tutti

Tra i regali del prossimo fascicolo un assembler per tutti i Commodore

Un facile approccio al linguaggio "naturale" del vostro computer.

di Alessandro de Simone

IL sistema di numerazione usato nei microprocessori è quello binario puro e pertanto difficilissimo da rappresentare, ma con un piccolo artificio si può semplificare tutto notevolmente ricorrendo al sistema di numerazione esadecimale consistente di sedici simboli:

0 1 2 3 4 5 6 7 8 9 A B C D E F

Per esempio, il numero 12 in base dieci è 1100 in binario puro e semplicemente C in esadecimale (fig. 1).

Poichè (fig. 2) la memoria del computer è una successione di gruppi di 8 bit ciascuno, detti parole o bytes, noi dovremmo scrivere un programma in binario puro, proprio come indicato in figura 2; figurarsi la confusione che ne verrebbe fuori per un programma di un centinaio di parole.

Se però dividiamo idealmente con una linea tratteggiata la 'striscia' otteniamo due gruppi di quattro bit ciascuno per ogni parola che, tradotti in codice esadecimale, sono molto più facili da scrivere in memoria e da controllare in seguito. Le prime tre parole in figura 2 anzichè 10101001, 11111111, 10001101 si scrivono A9 FF 8D ecc. Naturalmente c'è bisogno di un programma che traduca da codice esadecimale, scritto mediante la tastiera del PET utilizzando cifre e lettere, in binario puro, dato che il computer ragiona solamente con questo sistema.

Tale programma, piuttosto sofisticato, è residente nei PET dotati di nuove ROM e richiamabile battendo l'istruzione SYS(4).

Diamo comunque per scontato che il lettore sappia usare tale programma.

L'esempio considerato in figura 2 è un semplicissimo programma in L.M. (linguaggio macchina). Prima di commentarlo definiamo l'accumulatore: questo è un registro di otto bit interno alla CPU (6502) attraverso il quale viene elaborata una grande quantità di informazioni presenti in un programma. Infatti quando facciamo partire il programma

Numero decim.	Numero esadec.	Binario puro
0	0	0 0 0 0
1	1	0 0 0 1
2	2	0 0 1 0
3	3	0 0 1 1
4	4	0 1 0 0
5	5	0 1 0 1
6	6	0 1 1 0
7	7	0 1 1 1
8	8	1 0 0 0
9	9	1 0 0 1
10	A	1 0 1 0
11	B	1 0 1 1
12	C	1 1 0 0
13	D	1 1 0 1
14	E	1 1 1 0
15	F	1 1 1 1

Figura 1

Indir. base 10	Indir. Base 16 Zona alta	Zona bassa	Dato contenuto nell'indirizzo binario	esadec.
826	03	3A	1010 1001	A9
827	03	3B	1111 1111	FF
828	03	3C	1000 1101	8D
829	03	3D	0000 0000	00
830	03	3E	1000 0000	80
831	03	3F	0110 0000	60
832	03	40	....	....
833	03	41	....	....
.....	...	....	....	....
.....	...	....		

Figura 2

A9 FF 8D 00 80 60 la CPU carica il dato A9, e poichè le abbiamo detto che la locazione 033A contiene il primo dato da elaborare, capisce che A9 è un'istruzione ed esattamente, leggendo A9, carica nell'accumulatore la parola contenuta nella successiva locazione (033B), cioè FF.

Tale concetto è di fondamentale importanza: la CPU non può sapere se una certa locazione di memoria contiene un dato oppure un'istruzione. Sa soltanto (perchè è microprogrammata in tale modo) che la parola di partenza del programma deve considerarla come un'istruzione e continua l'elaborazione con questo presupposto.

Se infatti facciamo girare il programma da 033B anzichè da 033A, la CPU "pensa" che FF sia l'istruzione da eseguire e va subito in 'tilt' perchè non esiste questa istruzione fra le 151 istruzioni del 6502.

L'istruzione A9 è un'istruzione a due byte, cioè quando la CPU legge A9, sa che la locazione di memoria successiva contiene un dato e non un'istruzione, caricato, allora, il dato FF nell'accumulatore, leggerà il contenuto di 033C, che è 8D, istruzione, questa volta, a tre byte.

Un'istruzione a due byte viene conteggiata nel modo seguente: comando + dato; così similmente 8D, come tutte le istruzioni a tre byte, è costituita da: comando + dato + dato.

Incontrando 8D, la 6502 eseguirà il comando trascrivendo il dato presente nell'accumulatore, nella locazione di memoria il cui indirizzo è rappresentato, in esadecimale, dai dati contenuti nelle due locazioni di memoria successive a quelle contenente il comando 8D.

In parole un po' semplici la CPU scrive FF nella locazione, il cui indirizzo è 8000 esadecimale. Attenzione: il primo byte successivo a 8D contiene la parte bassa dell'indirizzo (LA= low address) mentre la terza parola contiene la parte alta (HA= high address).

Dato che vogliamo scrivere il dato FF, che è in accumulatore, nella locazione 8000 scriviamo 8D 00 80 e non 8D 80 00. Tale modo di fare è una caratteristica di tutte le istruzioni a tre byte che incontreremo prima di scrivere la LA poi la HA.

Dopo aver eseguito l'istruzione 8D, la CPU incontra l'istruzione 60 (istr. a un solo byte) equivalente al RETURN. Poichè non è stata chiamata alcuna subroutine si ritorna automaticamente al BASIC uscendo in tal modo dal L.M. Su tale concetto torneremo più avanti. Per ora basti sapere che se in un programma in L.M. c'è un RETURN (60)... in più, si ritorna al BASIC.

Vediamo ora a che cosa serve il programma appena esaminato. La memoria di schermo è rappresentata da 1000 locazioni di memoria (25 righe per 40 colonne) a partire dall'indirizzo 80 00. Scrivendo allora FF in 80 00, apparirà il simbolo "▚" nella prima locazione della memoria di schermo (in alto a sinistra). Modificando opportunamente il contenuto di 0800 e facendo rigirare il programma in L.M., faremo apparire nell'angolo in alto a sinistra tutti i simboli grafici del PET. Si potrebbe obiettare che il medesimo risultato si poteva ottenere semplicemente mediante il BASIC.

```
10 PRINT "▚"
```

Ciò è vero, però c'è da fare una prima considerazione: per far apparire in L.M. il simbolo "▚", abbiamo utilizzato solamente 6 byte; con il BASIC invece occuperemmo ben 10 byte, senza contare che si è dovuto utilizzare l'interprete del BASIC impiegando molto tempo. La sostanziale differenza fra i due metodi si nota quando si vorrà riempire tutto lo schermo con il carattere desiderato. Con il BASIC scriveremmo, ad esempio:

```
10 A=1
20 B=1000
30 PRINT "[clr]"     REM CANCELLAZIONE
40 FOR I=A TO B               SCHERMO
50 PRINT "▚"
60 NEXT
70 END
```

Eseguendo il programma si può nettamente seguire il formarsi dei 1000 simboli.

Proviamo ora a caricare il programma in L.M. che segue:

Ind. part.	Dati
033A	A9 80 8D 48 03 A9 00 8D
0342	47 03 A9 FF 8D 00 80 EE
034A	47 03 D0 F6 AD 48 03 C9
0352	83 D0 01 60 EE 48 03 4C
035A	44 03 ... ...

e... SYS(826)... è sorprendente, vero? Ma vediamo come gira:

```
100 I = 826
110 READ A$: IF A$="-1" THEN END
120 A1$ = LEFT$(A$,1)
130 A2$ = RIGHT$(A$,1)
140 IF A1$<"A" THEN 160
150 A1$ = STR$(ASC(A1$)-55)
160 IF A2$<"A" THEN 180
170 A2$ = STR$(ASC(A2$)-55)
180 A = VAL(A1$) * 16 + VAL(A2$)
200 POKE I,A: I = I+1
220 GOTO 110
230 DATA A9,80,8D,48,03,A9,00,8D
240 DATA 47,03,A9,FF,8D,00,80,EE
250 DATA 47,03,D0,F6,AD,48,03,C9
260 DATA 83,D0,01,60,EE,48,03,4C
270 DATA 44,03,-1
```

REMarks

033A A9 80 Nell'accumulatore viene scritto il numero 80 (in effetti viene scritto il numero binario puro 1000 0000 e non ripeteremo più questo concetto).

033C 8D 48 03 Il valore ora presente nell'accumulatore (80) viene riportato così come è nella locazione 03 48. Attenzione che nell'accumulatore è ancora presente 80.

033F A9 00 Nell'accumulatore viene ora scritto 00 cancellando il precedente 80; naturalmente ciò che è stato scritto in 03 48 non viene modificato.

0341 8D 47 03 Vedi istruzione 033C.

0344 A9 FF Vedi istruzione 033A oppure 033F.

0346 8D xx xx I due gruppi di xx stanno a significare che possiamo scrivere qualsiasi valore esadecimale; infatti quando in seguito faremo partire il programma, le istruzioni 033C e 0341 provvederanno a scrivere la parte alta e bassa dell'indirizzo; a questo punto dell'esecuzione del programma, appare in alto a sinistra il simbolo "▪", che vogliamo trascrivere anche nella seconda posizione, terza, ecc., fino a riempire tutto lo schermo. Dobbiamo pertanto fare in modo di impartire l'istruzione 8D 01 80 e poi 8D 02 80... fino a 8D FF 83. Per evitare di scrivere mille istruzioni (quante sono le celle dello schermo) utilizziamo un loop (fig. 3).

INDIR. ESA.	OP. COD.	MNEMONICO
033A	A9	LDA# 80
033B	80	
033C	8D	STA 0348
033D	48	
033E	03	
033F	A9	LDA# 00
0340	00	
0341	8D	STA 0347
0342	47	
0343	03	
0344	A9	LDA# FF
0345	FF	
0346	8D	STA 8000
0347	00	
0348	80	
0349	EE	INC 0347
034A	47	
034B	03	
034C	D0	BNE
034D	F6	
034E	AD	LDA 0348
034F	48	
0350	03	
0351	C9	CMP# 83
0352	83	
0353	D0	BNE
0354	01	
0355	60	RTS
0356	EE	INC 0348
0357	48	
0358	03	
0359	4C	JMP 0344
035A	44	
035B	03	

Fig. 3

0349 EE 47 03 Questa istruzione (EE) aumenta di una unità il contenuto della locazione, il cui indirizzo è 0347: da 00 passa a 01, oppure quando sarà contenuto A8, passerà ad A9 oppure (importante) da FF a 00.

034C D0 F6 L'istruzione D0 è un'istruzione a due byte che fa parte delle così dette istruzioni di BRANCH (salto condizionato). Arrivati a questo punto, il proseguimento dell'elaborazione dipende dal risultato dell'ultima operazione effettuata. Nel caso specifico, se EE 47 03 fornisce un valore uguale a 00, l'istruzione D0 viene ignorata ed il programma prosegue con l'istruzione successiva (034E). In caso contrario la CPU legge il primo bit del secondo byte dell'istruzione (fig. 4:

Fig. 4

1111 0110)e, se esso è 0,eseguirà un salto in avanti, mentre se è 1 (nostro caso) farà un salto indietro. Per salto si intende il numero di parole (e non istruzioni) che bisogna escludere per rintracciare la successiva istruzione da eseguire. Nel nostro caso, la CPU esegue un conteggio alla rovescia partendo dall'istruzione successiva a D0 (034E), definendola come FF. In seguito va all'indietro fino a che arriva al numero contenuto nel secondo byte di D0, cioè F6, corrispondente all'ind. 0345.

Il programma, allora, continuerà ad essere eseguito dalla parola successiva (sempre all'indietro) a 0345, e cioè 0344 (A9). Bisogna pertanto prestare la massima attenzione al secondo byte delle istruzioni di BRANCH perchè, in caso di errore, potremmo far continuare il programma da un dato anzichè da un'istruzione oppure da un'istruzione indesiderata. E' ovvio che nelle prossime puntate amplieremo questi concetti. Notiamo facilmente che il blocco di istruzioni 0344-034D sarà eseguito 255 volte fino a che cioè 0347 non conterrà 00, consentendo alla CPU di ignorare D0 F6. Ogni volta che il blocco viene eseguito, lo schermo viene riempito di 256 simboli di "▪▪". Poichè lo schermo è di 1000 caratteri, il blocco deve essere eseguito quasi quattro volte. Vediamo come:

034E AD 48 03 carica in accumulatore il dato contenuto in 0348.

0351 C9 83 Questa istruzione a due byte compara, mediante sottrazione, l'accumulatore con il numero esa. 83; se i due numeri risultano uguali fornirà 00 come risultato (attenzione: l'accumulatore non viene comunque 'modificato' nè tantomeno il numero esa. 83).

0353 D0 01 In questo caso, a differenza di 034C, il primo bit del secondo byte è 0 (fig. 4) e pertanto nel caso in cui il risultato della comparazione 0351 sarà uguale a 00, sarà saltata una parola ed il programma continuerà da 0356.

0355 60 Return subroutine: si esce dal L.M. e si ritorna al BASIC.

0356 EE 48 03 A questa istruzione (vedi 0349) si giunge se il risultato della comparazione 0351 indica che la HA di 0346 (cioè 8D 47 80) non è ancora passato al valore 83, cioè se siamo ancora nella memoria di schermo.

R i e p i l o g a n d o, il blocco 0344-034D viene eseguito 255 volte mentre 0344-0359 viene eseguito 4 volte prima di ritornare al BASIC.

0359 4C 44 03 Istruzione di salto incondizionato: la continuazione del programma passa all'istruzione contenuta in 0344.

```
100 SYS (826)
110 GET A$: IF A$="" THEN 110
120 A= ASC(A$): POKE 837,A
130 GOTO 100
```

Il lettore potrà provare la validità di questa routine in linguaggio macchina, in precedenza caricata, con il programmino qui sopra suggerito. ▪

Guida mercato Commodore

Prodotto	Prezzo (IVA esclusa)

VIC - 20

Home Computer Vic 20 da 5K 199.000
Unità di espansione 295.000
Modulo di espansione 135.000
Cartuccia da 3K di memoria 66.000
Cartuccia da 8K di memoria 98.000
Cartuccia da 16K di memoria 172.000
Cartuccia Vic rel 95.000
Permette di controllare il funzionamento di allarmi antifurto, porte automatiche, telefoni, trasmittenti ed apparecchi similari.
Vic switch 225.000
Possono essere collegati fino a 16 VIC 20 con un floppy e una stampante (distanza massima 1500 mt.).
Interfaccia IEEE 488 175.000
Interfaccia centronics 115.000
RS232-C adapter 75.000
RS132-C adapter 75.000

Commodore 64

CPU 64K RAM 625.000
C 64 Executive 1.950.000
Sistema operativo CP/M 125.000
Consente di programmare il Commodore 64 in linguaggio CP/M, il più utilizzato sui Personal Computers. Permette inoltre di accedere alla enorme libreria di Software applicativi CP/M.
Pet speed 95.000
Compilatore basic che aumenta la velocità di esecuzione dei programmi di circa 40 volte.

Accessori per Vic e Commodore 64

Stampante plotter a colori 375.000
80 caratteri, per linea, 4 colori, alla risoluzione di 0,2 mm per passo.
Unità stampante 450.000
Stampa velocemente su carta normale quanto appare sul video: programmi, lettere, dati, grafici.
Unità stampante 595.000
Stampante 80 colonne, bidirezionale, 60 CPS, spaziatura programmabile, trazione a frizione o a trattore.
Registratore dedicato 120.000
Per memorizzare facilmente programmi e dati su normali cassette magnetiche.
Floppy disk drive 585.000
Veloce unità di memoria di massa ad alta capacità. Può immagazzinare fino a 170.000 caratteri su ogni singolo disco.
Monitor monocromatico 285.000
A fosfori verdi 12".

Per giocare

Comando per giochi (Joystick) 13.500
Permette di muoversi in tutte le direzioni, di iniziare i vari giochi di movimento e di "sparare".
Comando a manopola per giochi (Paddle) 22.500
Adatto per i giochi a 2 persone, esegue movimenti in orizzontale e verticale,

Commodore 4000

CPU 16K RAM 1.285.000
18K ROM, BASIC 4.0 residente, video 40 colonne per 25 righe, tastiera semigrafica.
CPU 32K RAM 1.385.000
18K ROM, BASIC 4.0 residente, video 40 colonne per 25 righe, tastiera semigrafica.

Commodore 9000

Doppia CPU 134K RAM 2.350.000
Micro Main Frame Computer a doppia CPU (6502 - 6809) compatibile con tutte le periferiche Commodore della serie 8000. Include 5 linguaggi di programmazione. (COBOL, FORTRAN, TCL PASCAL, UCSD PASCAL, APL).

Commodore 8000

CPU 32K RAM 1.675.000
18K ROM, Basic 4.0 residente, video orientabile e basculante 80 colonne per 25 righe, tastiera commerciale separata.
CPU 96K RAM 2.150.000
18K ROM, Basic 4.0 residente, video orientabile e basculante 80 colonne per 25 righe, tastiera commerciale separata. Include sistema operativo PM/96.

Commodore 600

Indicato per applicazioni industriali, collegamento a strumentazione, controllo numerico, ecc. Utilizza monitor in commercio.
CPU 128K RAM 2.150.000
CPU 128K RAM espandibile internamente a 256K ed esternamente a 960K, interfaccia RS232C, IEEE 488, porta utente a 8 Bit. Compatibile con tutte le periferiche Commodore della serie professionale.
CPU 256K RAM 2.550.000
CPU 256K RAM espandibili esternamente a 960K. Caratteristiche uguali al Mod. 610.
Monitor 285.000
Monocromatico a fosfori verdi, 12".

Commodore 700

CPU 128K RAM 2.850.000
CPU 128K RAM espandibili internamente a 256K ed esternamente a 960K. Video orientabile e basculante 80 colonne per 25 righe. Compatibile con tutte le periferiche Commodore delle serie professionali.
CPU 256K RAM 3.250.000

VIC 20

—SUPPORTO: CARTRIDGE—

6502 PROFESSIONAL DEVELOPMENT SYSTEM
Un sistema di sviluppo del 6502 su cassetta che traduce dall'assembler al linguaggio macchina in uno o due passi con lo standard mnemonico mos.

TF/9316-00 L. 60.000

TORG
Torg è un gioco di abilità ed avventura. Sei intrappolato in una griglia, devi cercare di evadere e di distruggere orde di mostri che ti attaccano per annientarti.

TF/9316-01 L. 36.000

COCO 2
Un fantastico Computer Games di tipo educativo, che ti permetterà di creare da solo i tuoi giochi. Coco 2 ti insegna la teoria dei giochi, la grafica dei computers, i segni della programmazione e il disegno logico. Coco 2 non richiede alcuna precedente esperienza di programmazione. È richiesta l'espansione 16 K.

TF/9316-03 L. 84.000

VIC FORTH
Un nuovo linguaggio di sicuro interesse che unisce la facilità del Basic alla velocità dell'assembler. Un ricco manuale a corredo.

TF/9316-20 L. 114.000

HESMON
Hesmon è un monito linguaggio macchina del 6502 con l'aggiunta di 30 comandi di utilità. Un potente accessorio con molte caratteristiche non reperibili sugli altri monitor.

TF/9316-22 L. 84.000

TURTLE GRAPHICS
Turtle Graphics è un modo eccittante e divertente per introdurre un principiante di concetti di programmazione. Un sistema facile per apprendere il linguaggio dei computer con più di 30 comandi a disposizione.

TF/9316-24 L. 84.000

HES WRITER
Un programma di Word Processing molto curato e con numerose opzioni tra le quali: Editing completo sullo schermo, giustificazione per centratura, numerazione delle pagine, ecc. compatibile con la stampante VC 1515 o Seikosha Gp 100 VC.

TF/9316-26 L. 84.000

AGGRESSOR
Come un valoroso pilota devi proteggere i depositi di benzina di Stellarium dagli attacchi delle navicelle di Zaurien. Stai attento alle mine spaziali e ai vari tipi di astronavi nemiche.

TF/9316-28 L. 84.000

SYNTHE SOUND
Trasformate il vostro Vic 20 in un incredibile sintetizzatore musicale. Questo packagg fornirà speciali effetti consentendovi di diventare un abile compositore musicale.

TF/9316-30 L. 114.000

GRID RUNNER
Accanitevi contro il nemico droido che si ammassa su «Grid» la stazione nucleare nell'orbita solare terrestre. Un gioco a livello arcade per grafica e sonoro. Livello di difficoltà a vostro piacimento per giocare utilizzare il joystick.

TF/9316-32 L. 84.000

SHAMUS
Alla fine rimarrai tu o l'Ombra!?! solo tu puoi porre fine al regno del terrore dell'Ombra maledetta! Devi combattere contro chiunque. 2 livelli con 20 stanze piene di Robots, che ti danno la caccia. La tua sola funzione è difendere il palazzo contro gli invasori. Dovrai essere veloce e spietato!

TF/9316-34 L. 84.000

PROTECTOR
Le orde della città di Fraxullan stanno attaccando la tua città e catturando gli abitanti. Tu devi aiutare la popolazione insieme, trasformandola nella Città della Nuova Speranza, sull'altro lato del vulcano; ma i sopravvissuti non sono ancora salvi. Ogni persona dovrà essere nuovamente trasportata nella fortezza di Verdann prima che il vulcano esploda, travolgendoli tutti!!!

TF/9316-36 L. 90.000

PREDATOR
Fiero come un'aquila, tu voli nel cielo, difendendo il tuo spazio dagli attacchi di nemici pennuti. Mano a mano che sconfiggi i tuoi avversari, avanzi lungo la scala di 99 livelli di difficoltà. Come un'aquila, tu aspiri a raggiungere altezze sempre più grandi. Un gioco per tutte le età. Realmente nuovo e differente dai tradizionali giochi.

TF/9316-40 L. 84.000

COMMODORE 64

—SUPPORTO: CARTRIDGE—

HESMON 64
Hesmon è un monitor linguaggio macchina per il 6502/6510 con un mini assembler, indispensabile per tutti coloro che vogliono programmare in linguaggio assembler.

TF/9200-00 L. 84.000

RETRO BALL
L'emozionante e divertente gioco dell'Hockey su ghiaccio riproposto ora sul computer. Possibilità di giocare fra 2 giocatori o contro il computer.

TF/9200-02 L. 84.000

TURTLE GRAPHICS
Turtle Graphics è un modo eccittante e divertente per introdurre un principiante di concetti di programmazione. Un sistema facile per apprendere il linguaggio dei computer con più di 30 comandi a disposizione.

TF/9200-04 L. 120.000

HES WRITER
Un programma di Word Processing molto curato e con numerose opzioni tra le quali: Editing completo sullo schermo, giustificazione per centratura, numerazione delle pagine, ecc. compatibile con la stampante VC 1515 o Seikosha Gp 100 VC.

TF/9200-06 L. 90.000

GRID RUNNER
Accanitevi contro il nemico droido che si ammassa su «Grid» la stazione nucleare nell'orbita solare terrestre. Un gioco a livello arcade per grafica e sonoro. Livello di difficoltà a vostro piacimento per giocare utilizzare il joystick.

TF/9200-08 L. 84.000

COCO 2
Un fantastico Computer Games di tipo educativo, che ti permetterà di creare da solo i tuoi giochi. Coco 2 ti insegna la teoria dei giochi, la grafica dei computers, i segni della programmazione e il disegno logico. Coco 2 non richiede alcuna precedente esperienza di programmazione. È richiesta l'espansione 16 K.

TF/9205-00 L. 90.000

COCO
Eccezionale computer games di tipo educativo, che insegna le tecniche fondamentali del computer, il linguaggio Basic, come risolvere i problemi scomponendo ogni programma in semplici pezzi e simulando l'effetto di ogni istruzione. Coco ti permette di capire come lavora un computer.

TF/9205-02 L. 100.000

6502 PROFESSIONAL DEVELOPMENT SYSTEM
Un sistema di sviluppo del 6502 su cassetta che traduce dall'assembler al linguaggio macchina in uno o due passi con lo standard mnemonico mos.

TF/9205-04 L. 60.000

BENJI'S SPACE RESCUE
Assieme al cane Benji', stella di molti sceneggiati televisi americani, intrapprenderai un viaggio attorno al sistema solare, alla ricerca di alcuni scienziati rapiti. Scoprirai le meraviglie del cosmo, e combatterai contro alieni, insieme al fedele Benj'i. Dovrai mettercela veramente tutta per scoprire dove sono tenuti prigionieri gli scienziati.
Ce la farai?

TF/9205-06 L. 90.000

ZX 81

—SUPPORTO: CASSETTA—

BUDGET MASTER 1000 16 K
Uno splendido programma per la gestione casalinga. È possibile creare una serie di categorie specifiche e visualizzare costantemente la situazione del mese in corso, dell'anno, ecc.

TF/0106-00 L. 36.000

RIVERSI 1000 16 K
Un famoso gioco del diciannovesimo secolo, anche conosciuto come Othello, presentato per lo ZX 81 in una forma molto sofisticata e ricca di opzioni.

TF/0106-02 L. 36.000

SABOTAGE 16 K
Un'intricata avventura per giocatori esperti e dai riflessi veloci.

TF/0106-04 L. 30.000

GULPER 16 K
Un gioco di qualità arcade di caccia, catture ed evasioni. 15 selezioni di area gioco 9 livelli e 9 velocità lo rendono veramente completo ed entusiasmante.

TF/0106-06 L. 30.000

è distribuito da

Casella Postale 10488 MI

espande all'infinito la tua esperienza

software a misura d'uomo

CPU 256K RAM espandibili esternamente a 960K. Video orientabile basculante 80 colonne per 25 righe. Compatibile con tutte le periferiche Commodore della serie professionali.

Dischi

Floppy disk drive 650.000
Unità di memoria di massa ad alta velocità. Capacità 170KB. Drive singolo.

Floppy disk drive 1.900.000
Unità di memoria di massa ad alta velocità. Capacità 343KB. Drive doppio.

Floppy disk drive 2.125.000
Drive doppio 1M byte in linea.

Floppy disk drive 2.450.000
Drive doppio, doppia faccia, doppia densità 2M byte in linea.

Floppy disk drive 2.600.000
Drive doppio, doppia faccia, doppia densità, 2M byte in linea.

Hard disk 6.200.000
Tecnologia Winchester, .5M byte in linea.

Hard disk 6.700.000
Tecnologia Winchester, 7.5M byte in linea.

Stampanti

Stampante 625.000
Bidirezionale ad aghi, 60 CPS, 80 colonne.

Stampante 1.195.000
Stampante ad aghi 150 CPS, 132 colonne, bidirezionale, trascinamento a trattore.

Stampante 2.950.000
Stampante a margherita, 40 CPS, 136 colonne passo pica, 163 colonne passo élite, bidirezionale, utilizzabile anche con carta da bollo, trascinamento a frizione o a trattore.

Accessori

Microprocessore 32K RAM 2.500.000
Per connettere, in rete fino a 16 CPU RS232, IEEE 488, centronics. Il prezzo include (configurazione minima): controller, terminator, 3 moduli, cavi, cavo IEEE/PET (per la versione SK).

Singolo modulo addizionale: 325.000
1 modulo
1 cavo 6 piedi.

Nuovo sistema operativo 95.000
Per 8096SK o per 8032SK con B - 1 oppure con B - 2. Può gestire fino a 16 programmi residenti simultaneamente in memoria. Da a disposizione 26K per le variabili e 53K per i programmi. Potenzia inoltre il Basic con altri comandi.

64K RAM 575.000
Scheda di ampliamento memoria per 8032 e nuovo sistema operativo "PM 96"

CP/Maker 1.450.000
Incrementa la memoria interna di 64K RAM e permette l'uso di tutti i programmi CP/M. 8 bit disponibili. Compatibile con la serie 3000/4000/8000.

Scheda ad alta risoluzione grafica 720.000
Compatibile ai sistemi della serie 8000.

Cavo PET/IEEE 488 85.000

Cavo IEEE 488/IEEE 95.000

Accoppiatore acustico .595.000
300 baud/sec.

```
1800 PRINTE1$E3$" INIZ.";:INPUTA1$:L=LEN(A1$):A1=VAL(A1$)
1805 IFA1$="←"THENRETURN
1810 IFL>5THENPRINTE2$:GOSUB1000:GOTO1800
1820 PRINTE1$E3$" FIN.";:INPUTA1$:L=LEN(A1$):A2=VAL(A1$)
1825 IFA1$="←"THEN1800
1830 IFL>5 OR A2<=A1THENPRINTE2$:GOSUB1000:GOTO1820
1840 PRINTE1$A1" "A2;E6$:GOSUB1000:IFX$<>"S"THENRETURN
1850 PRINTE1$"NEXT. DISK":GOSUB1000:IFX$="←"THENRETURN
1860 REM IFX$="N"THENGOSUB1890:RETURN
1870 IFX$="D"THENGOSUB1900:RETURN
1880 GOTO1850
1890 REM INSERIRE QUI LA ROUTINE PER IL NASTRO
1900 PRINTE1$"NOME";:INPUTA1$:L=LEN(A1$)
1910 IFA1$="←"THENRETURN
1920 IFL>16THENRETURN
1925 A$="@0:"+A1$+",SEQ,WRITE"
1930 OPEN1,8,2,A$
1940 PRINT#1,A1
1950 FORI=A1TOA2:PRINT#1,PEEK(I):NEXT:PRINT#1,1000:CLOSE1:RETURN
1999 REM *** LETTURA DA DISCO ***
2000 PRINTE1$"NOME";:INPUTA1$:L=LEN(A1$)
2010 IFA1$="←"THENRETURN
2020 IFL>16THENRETURN
2030 A$="0:"+A1$+",SEQ,READ"
2040 OPEN1,8,2,A$
2050 INPUT#1,I
2060 INPUT#1,X:IFX>255THENCLOSE1:RETURN
2070 POKEI,X:I=I+1:GOTO2060
```